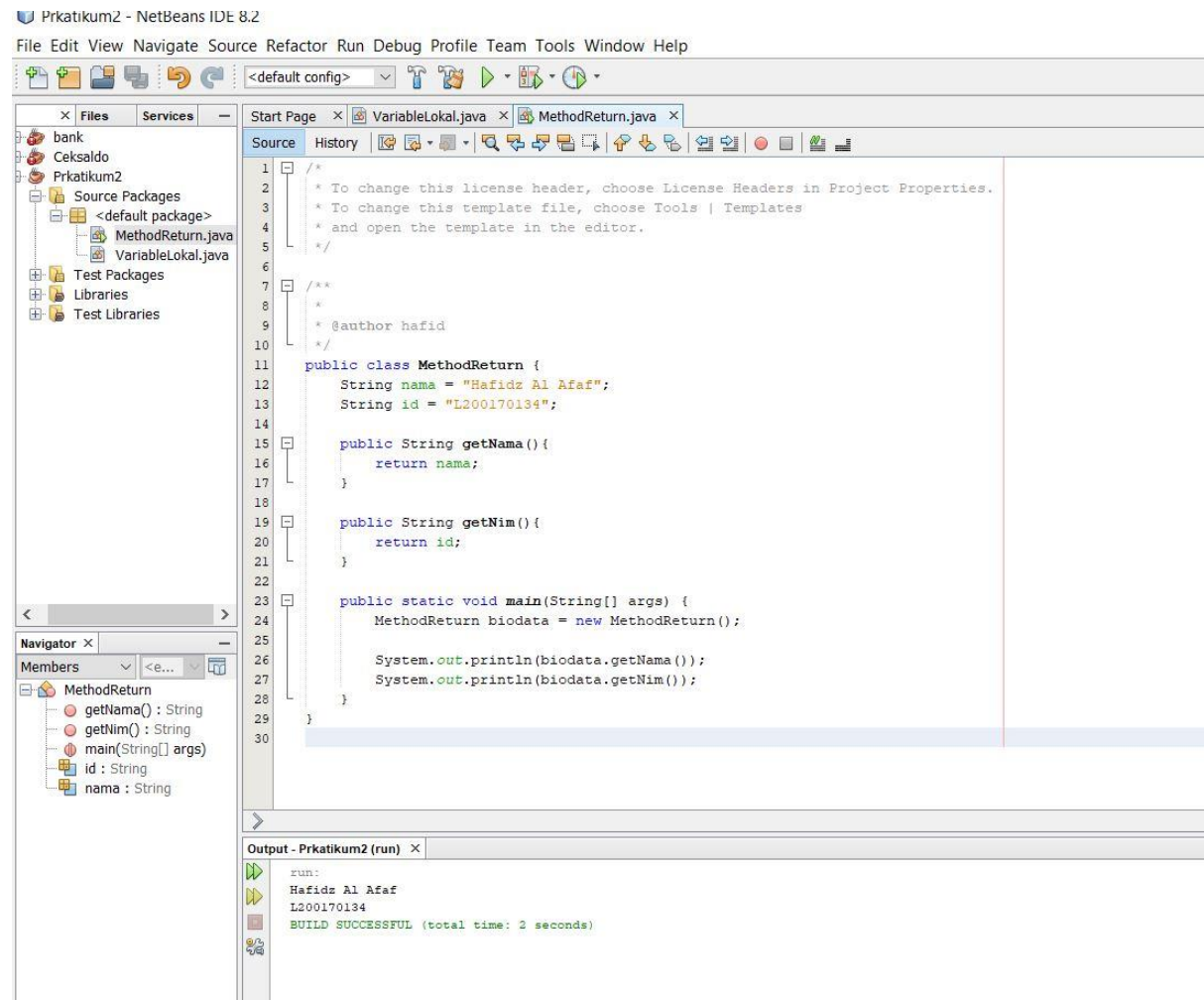


NAMA : HAFIDZ AL AFAF
NIM : L200170134
KELAS : C

MODUL 3

1.METHODRETURN



2. PEGAWAI

The screenshot displays an IDE interface with the following components:

- Top Menu Bar:** File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help.
- Toolbar:** Includes icons for file operations, running, and debugging.
- Left Panel (Project Explorer):** Shows a project structure with packages like 'bank', 'Ceksaldo', 'Prkatikum2', and 'Source Packages'. The 'Pegawai.java' file is selected under 'Source Packages'.
- Bottom Left Panel (nlp - Navigator):** Displays the 'Members' of the 'Pegawai' class, including methods like 'infoPegawai()', 'main()', 'setGaji()', 'setNama()', 'setNip()', and fields 'gaji', 'nama', and 'nip'.
- Main Editor:** Contains the source code for the 'Pegawai' class. The code defines a class with attributes 'nama' (String), 'nip' (int), and 'gaji' (double). It includes setter methods 'setNama()', 'setNip()', and 'setGaji()', an 'infoPegawai()' method for printing details, and a 'main()' method for program execution.
- Bottom Right Panel (Output - Prkatikum2 (run)):** Shows the execution output, displaying the details of five employees: Salsa, Rizki, Aini, Puspita, and Udin, each with their respective NIP and salary.

```
10 //
11 public class Pegawai {
12     String nama;
13     int nip;
14     double gaji;
15
16     public String setNama(String beriNama) {
17         return this.nama = beriNama;
18     }
19
20     public int setNip(int beriNip) {
21         return this.nip = beriNip;
22     }
23
24     public double setGaji(Double beriGaji) {
25         return this.gaji = beriGaji;
26     }
27
28     void infoPegawai() {
29         System.out.println(
30             "Nama: " + this.nama + "\n" +
31             "Nip: " + this.nip + "\n" +
32             "Gaji: Rp. " + this.gaji + "\n"
33         );
34     }
35
36     public static void main(String[] args) {
37         Pegawai datapegawail = new Pegawai();
38     }
39 }
```

Output - Prkatikum2 (run)

```
run:
Nama: Salsa
Nip: 113
Gaji: Rp. 2000000.0

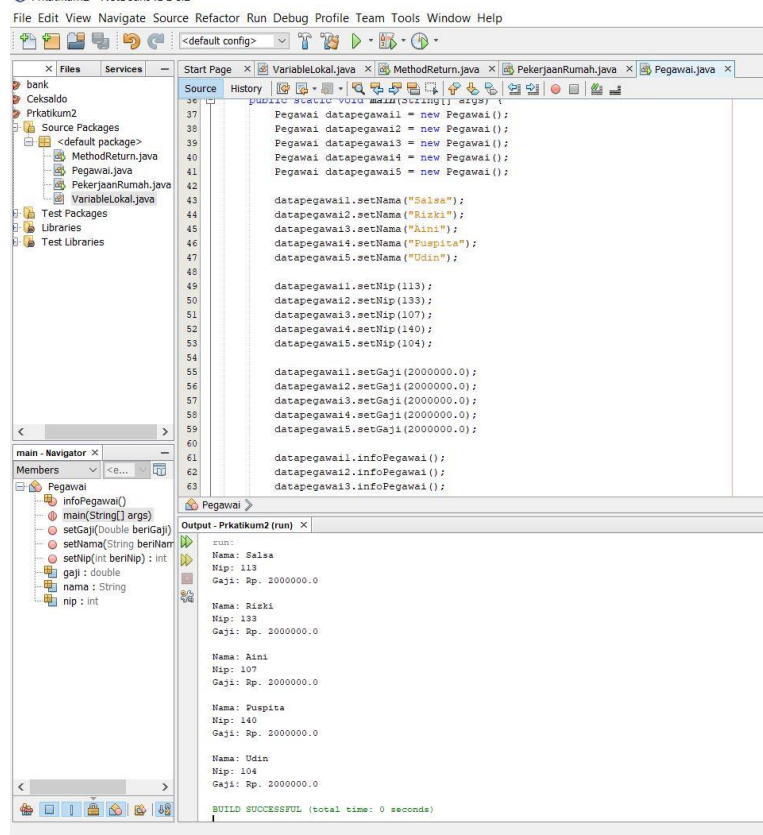
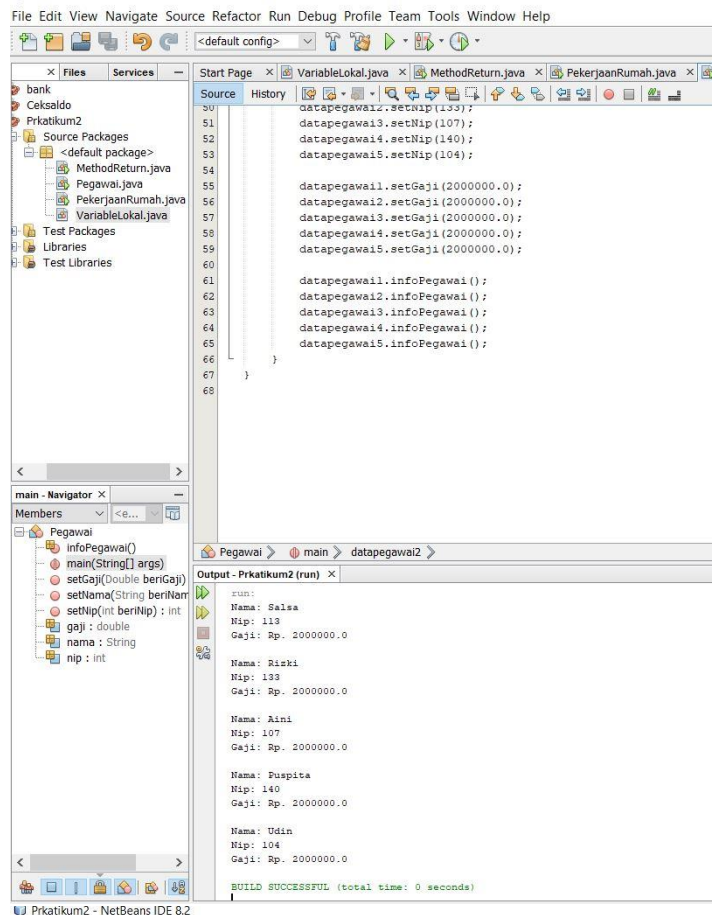
Nama: Rizki
Nip: 133
Gaji: Rp. 2000000.0

Nama: Aini
Nip: 107
Gaji: Rp. 2000000.0

Nama: Puspita
Nip: 140
Gaji: Rp. 2000000.0

Nama: Udin
Nip: 104
Gaji: Rp. 2000000.0

BUILD SUCCESSFUL (total time: 0 seconds)
```



PEKERJAAN RUMAH

The screenshot shows an IDE with the following components:

- Navigator:** Displays the project structure. The `PekerjaanRumah` class is selected, showing its methods: `main(String[] args)`, `setMID(int nilaiMID)`, `setTotal()`, `setTugas(int nilaiTugas)`, and `setUAS(int nilaiUAS)`. The variables `mid` (double), `nilaiMID` (int), `nilaiTugas` (int), `nilaiUAS` (int), `tugas` (double), and `uas` (double) are listed.
- Source Editor:** Contains the implementation of the `PekerjaanRumah` class. The code defines the class with attributes `nilaiMID`, `nilaiTugas`, and `nilaiUAS` of type `int`, and `mid`, `tugas`, and `uas` of type `double`. It implements the `setMID`, `setTugas`, and `setUAS` methods, each multiplying the input value by 1.0. The `setTotal` method prints the values and calculates the total as $\frac{(mid + tugas + uas)}{3}$. The `main` method creates a new `PekerjaanRumah` object and calls the `setTotal` method.
- Output - Pratikum2 (run):** Shows the execution results: `Nilai MID: 80`, `Nilai Tugas: 90`, `Nilai UAS: 80`, and `Total: 86.66666666666667`. The build is successful.

The screenshot shows the same IDE with the `main` method implemented in the `PekerjaanRumah` class. The `main` method now includes the following code:

```
public static void main(String[] args) {
    PekerjaanRumah nilai = new PekerjaanRumah();

    nilai.setMID(90);
    nilai.setTugas(90);
    nilai.setUAS(80);
    nilai.setTotal();
}
```

The `setTotal` method remains the same, printing the values and calculating the total. The `Output - Pratikum2 (run)` window shows the same results as the previous screenshot, confirming the successful execution of the program.