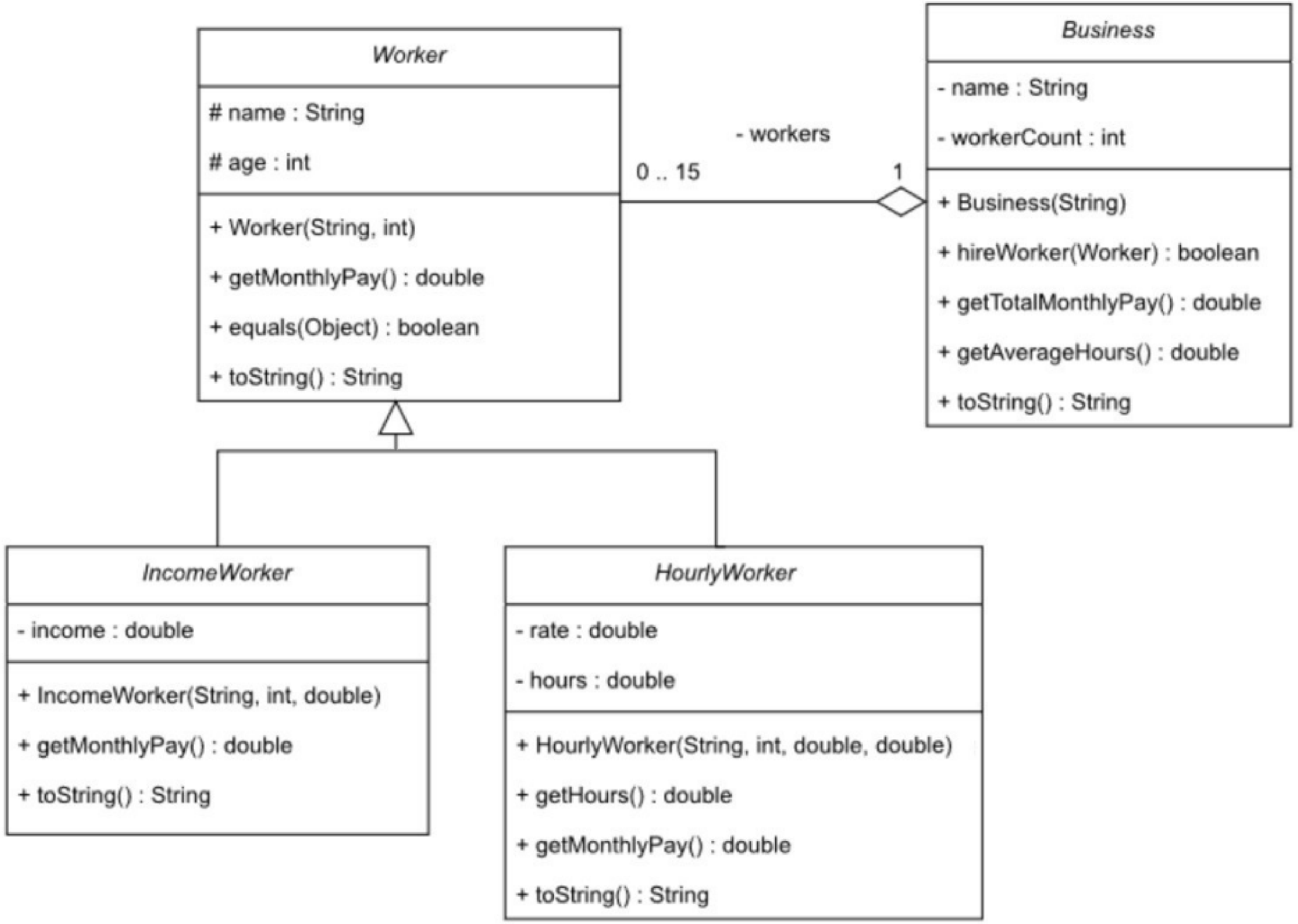
CSC 113

ABSTRACT

We want to write a program that manages the monthly worker payments for a business using the following UML diagram:



Class **Worker:**

- Instance Attributes:
  - ***name***: the name of the worker
  - ***age***: the age of the worker

- Methods:
  - ***Worker(name:String, age:int):*** constructor
  - ***equals(obj:Object):*** compares two workers based on their ***name*** and ***age*** and returns true if they are equivalent or false otherwise.
  - ***toString():*** returns a string representation of the worker in the following format:
    - **Worker: *name*, age: *age***
  - ***getMonthlyPay():*** returns the monthly pay for a worker:
    - IncomeWorker: the income paid in a month
    - HourlyWorker: the rate paid per hour of work in a month

Class **IncomeWorker:**

- Instance Attributes:
    - ***income***: yearly income

- Methods:
    - ***IncomeWorker(name:String, age:int, income:double):*** constructor
    - ***toString():*** returns a string representation of the worker in the following format:
        - **Worker: *name*, age: *age*, income*: income***

Class **HourlyWorker:**

- Instance Attributes:
    - ***rate***: the rate paid per hour
    - ***hours***: the number of hours worked in a month

- Methods:
    - ***HourlyWorker(name:String, age:int, rate:double, hours:double):*** constructor
    - ***getHours():*** returns ***hours***
    - ***toString():*** returns a string representation of the worker in the following format:
        - **Worker:** ***name*****, age:** ***age*****, rate:** ***rate*****, hours:** ***hours***

Class **Business:**

- Instance Attributes:
  - ***name:*** name of the business
  - ***workers:*** array of Worker objects
  - ***workerCount:*** number of Worker objects in ***workers***

- Methods:
  - ***Business(name:String):*** constructor
  - ***hireWorker(w:Worker):*** adds ***w*** to the first available space of ***workers*** array if there's space and the worker isn't in the array already and returns true. Otherwise, returns false.
  - ***getTotalMonthlyPay***(): returns the total monthly pay for all workers in the business
  - ***getAverageHours***(): returns the average hours for all hourly workers in the business
  - ***toString():*** returns a string representation of the worker in the following format:
    - **Business *name* (*workerCount*):**
      - **Worker: *name*, age: *age,* income*: income***
      - **Worker: *name*, age: *age,* rate*: rate,* hours*: hours***
      - **…**

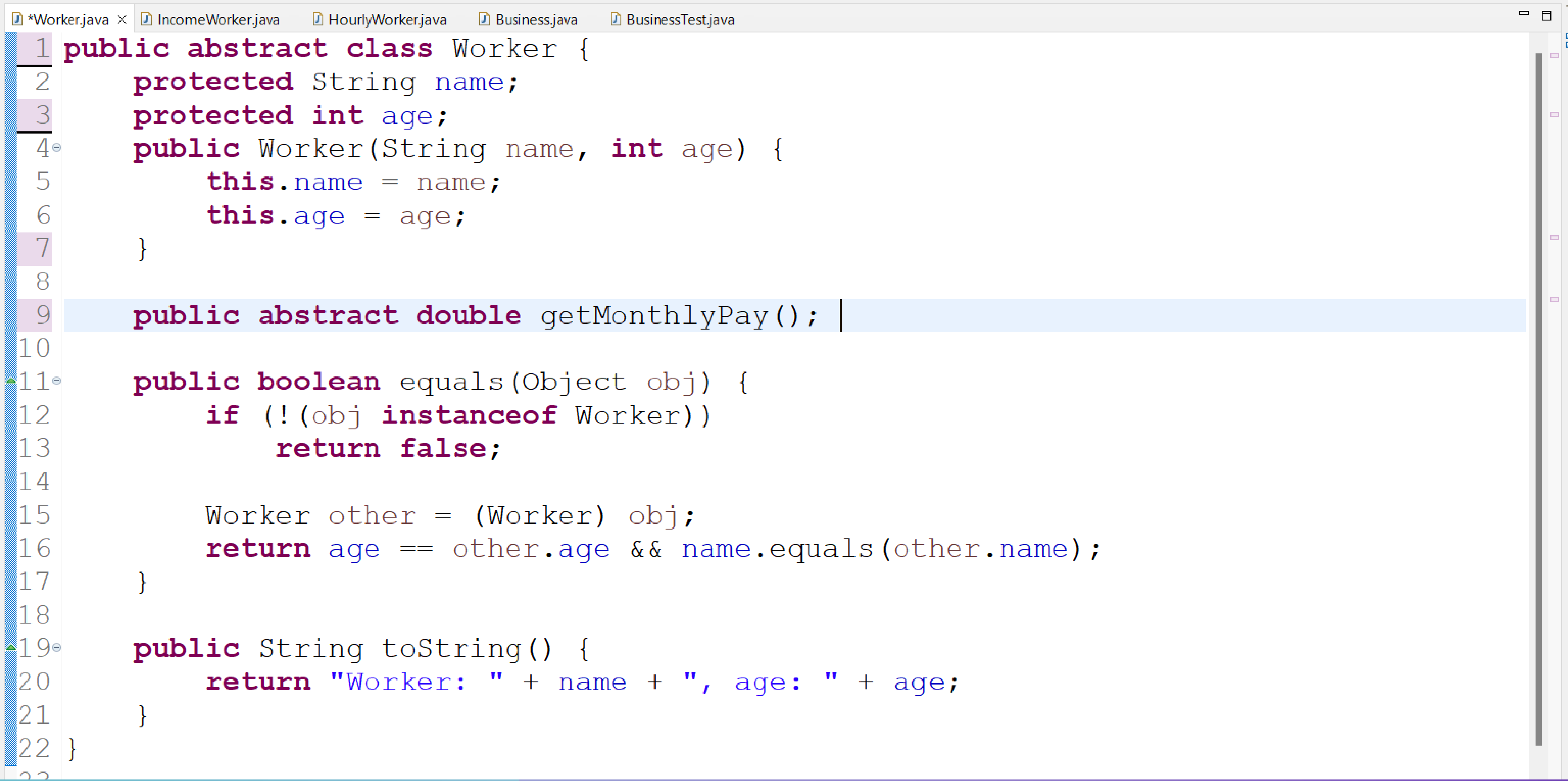
Exercise 1: Write classes **`Worker, IncomeWorker, HourlyWorker & Business.`**

Exercise 2: Write class **`BusinessTest`** that has a main method to test the functionalities of the classes:

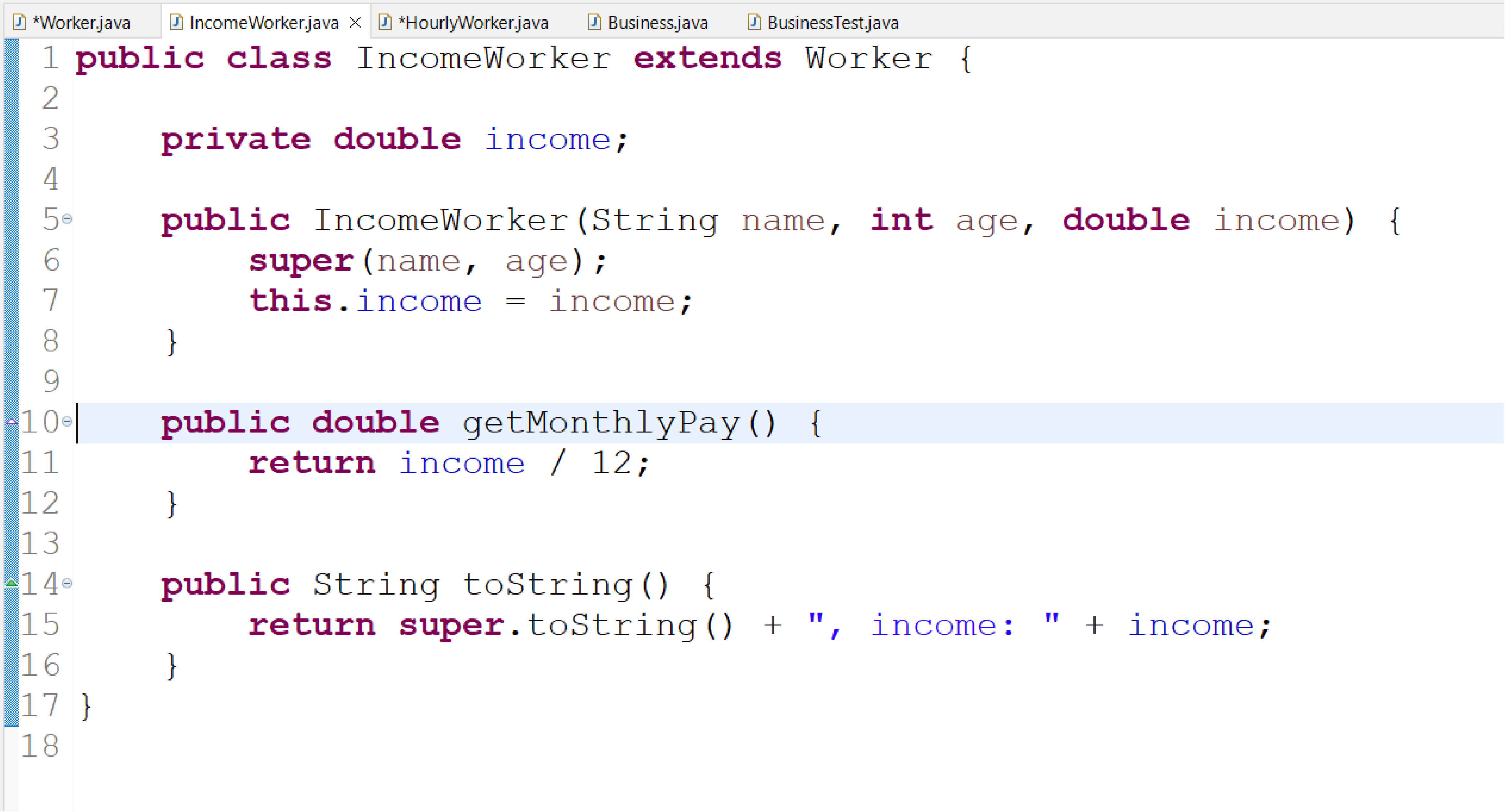
- Create a business
- Hire 4 workers (see sample run)
- Print the business
- Print the total monthly pay for the business
- Print the average hours for hourly workers in the business

Sample run:

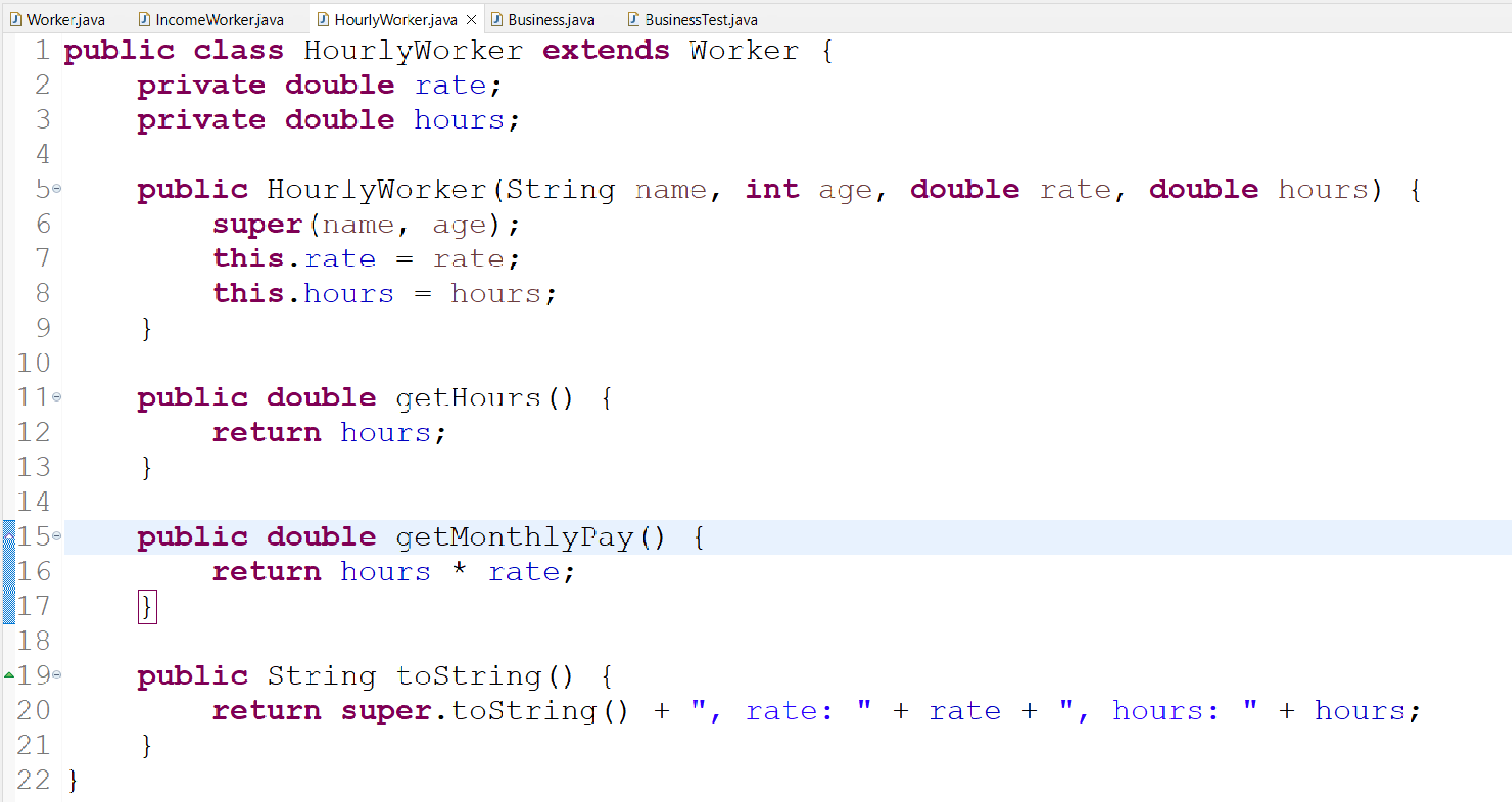
```
Business SICP (4):

Worker: Saleh, age: 23, income: 180000.0

Worker: Noor, age: 21, income: 120000.0

Worker: Jawaher, age: 25, rate: 80.0, hours: 160.0

Worker: Faisal, age: 22, rate: 60.0, hours: 200.0

Total monthly pay: 49800.0

Average hours: 180.0
```



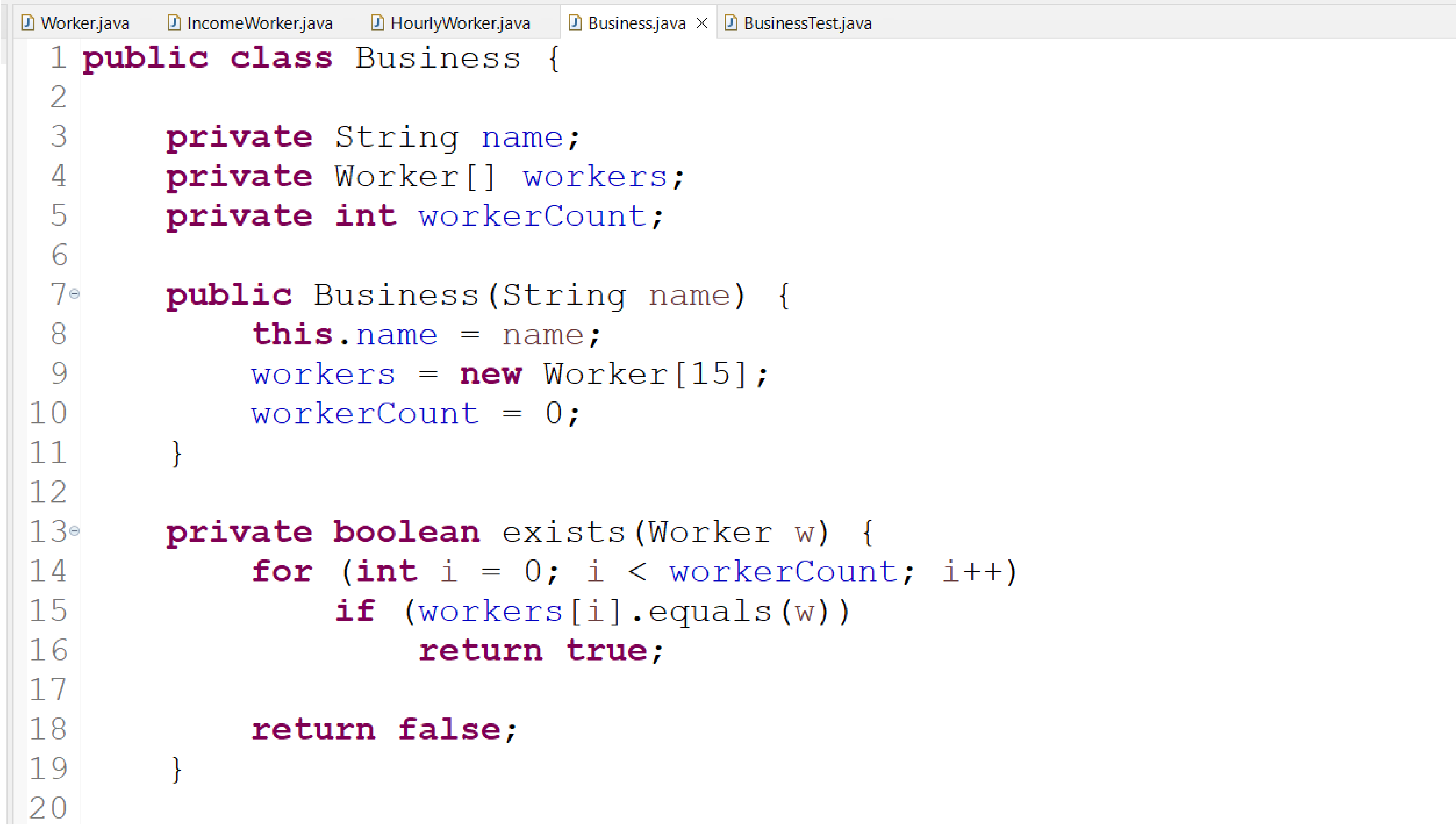
```java
public abstract class Worker {
    protected String name;
    protected int age;
    public Worker(String name, int age) {
        this.name = name;
        this.age = age;
    }

    public abstract double getMonthlyPay();

    public boolean equals(Object obj) {
        if (!(obj instanceof Worker))
            return false;

        Worker other = (Worker) obj;
        return age == other.age && name.equals(other.name);
    }

    public String toString() {
        return "Worker: " + name + ", age: " + age;
    }
}
```



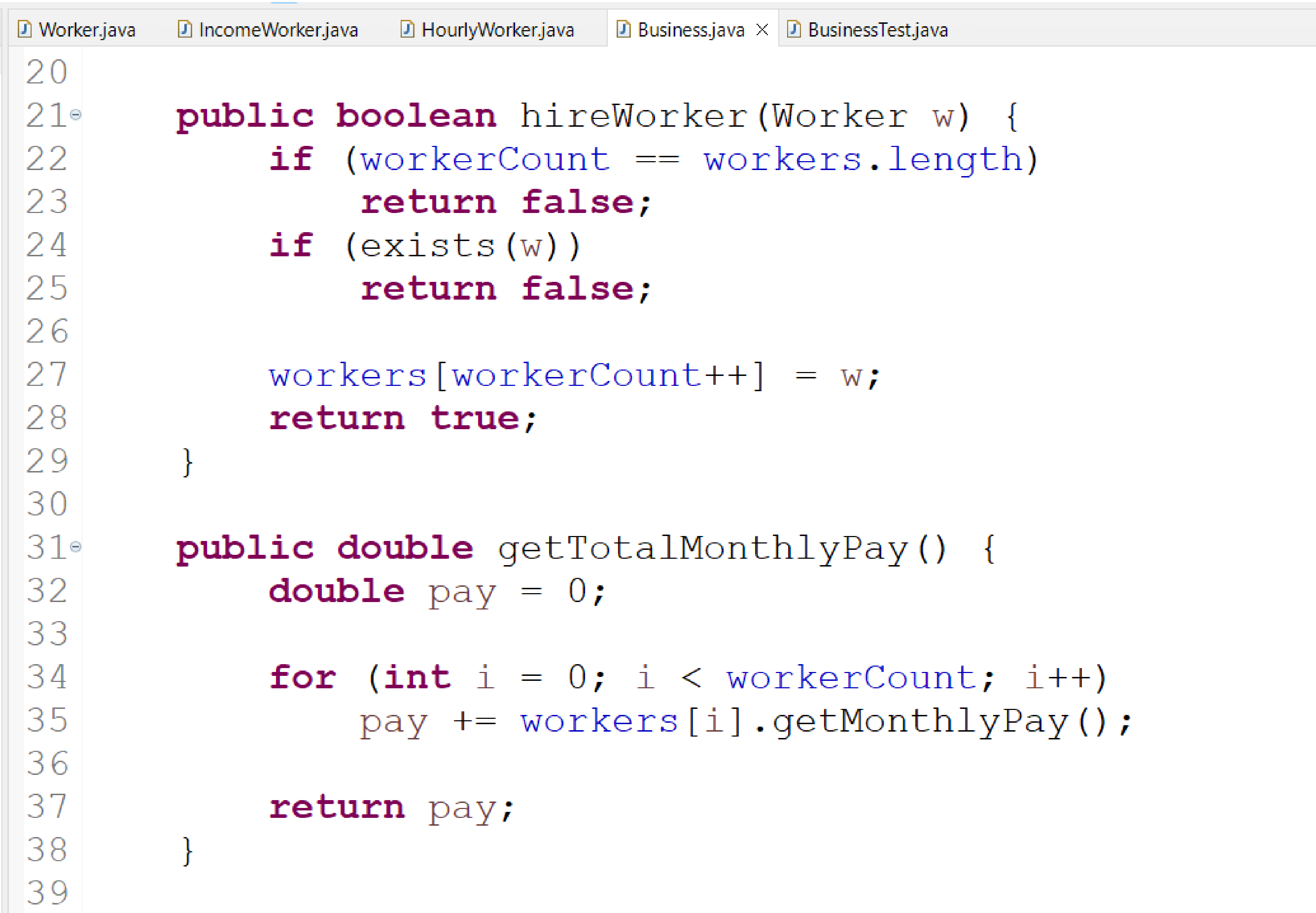
```java
public class IncomeWorker extends Worker {

    private double income;

    public IncomeWorker(String name, int age, double income) {
        super(name, age);
        this.income = income;
    }

    public double getMonthlyPay() {
        return income / 12;
    }

    public String toString() {
        return super.toString() + ", income: " + income;
    }
}

```



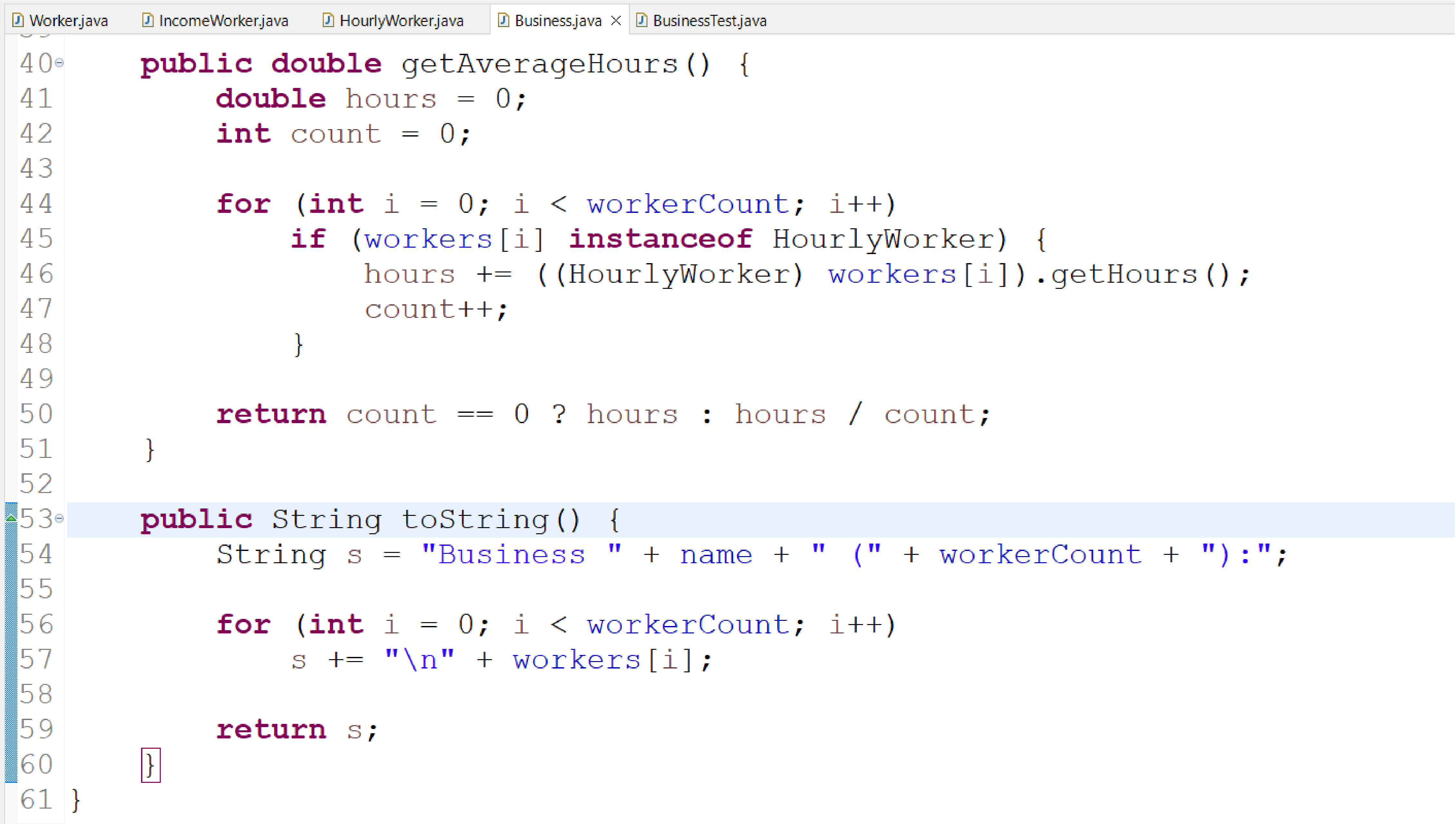
```java
public class HourlyWorker extends Worker {
    private double rate;
    private double hours;

    public HourlyWorker(String name, int age, double rate, double hours) {
        super(name, age);
        this.rate = rate;
        this.hours = hours;
    }

    public double getHours() {
        return hours;
    }

    public double getMonthlyPay() {
        return hours * rate;
    }

    public String toString() {
        return super.toString() + ", rate: " + rate + ", hours: " + hours;
    }
}
```



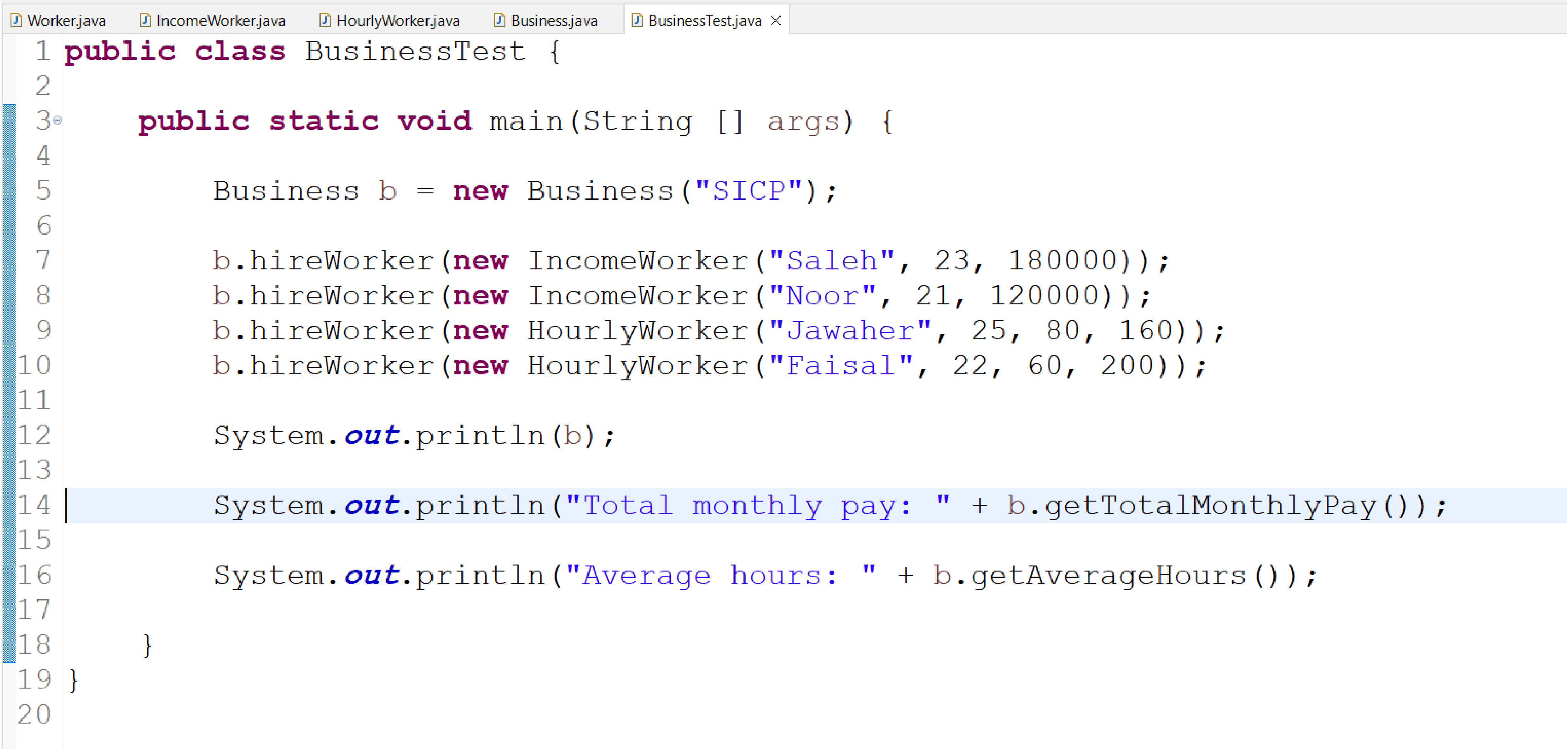
```java
public class Business {

    private String name;
    private Worker[] workers;
    private int workerCount;

    public Business(String name) {
        this.name = name;
        workers = new Worker[15];
        workerCount = 0;
    }

    private boolean exists(Worker w) {
        for (int i = 0; i < workerCount; i++)
            if (workers[i].equals(w))
                return true;

        return false;
    }

```



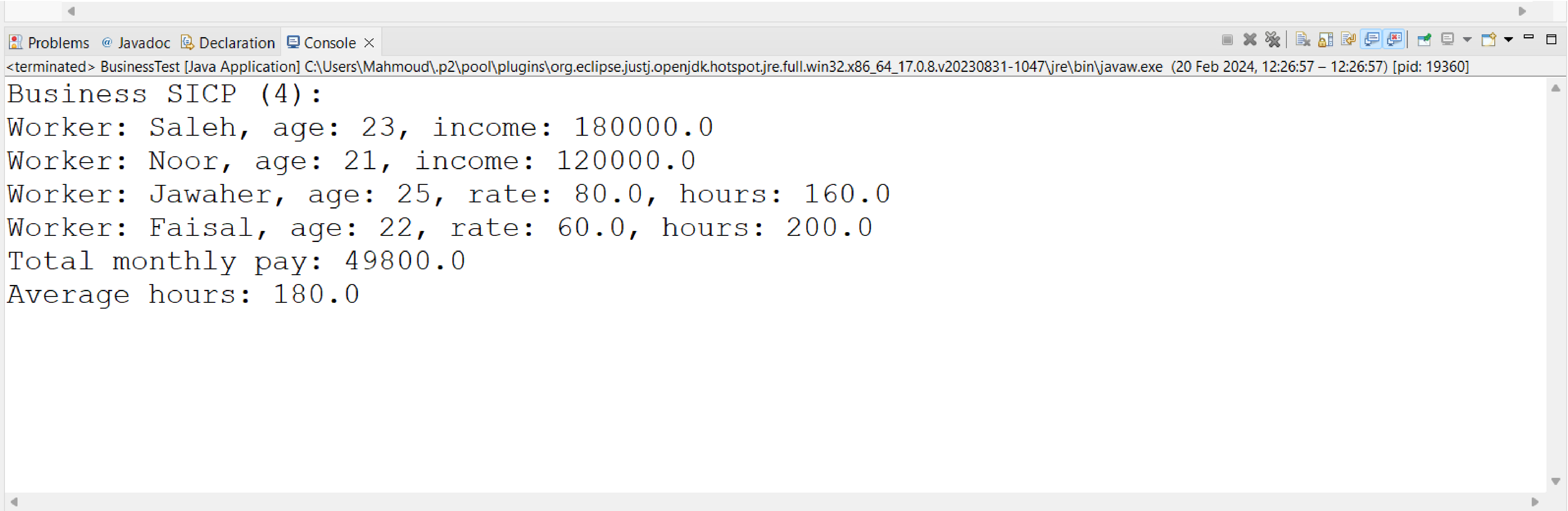
```java

    public boolean hireWorker(Worker w) {
        if (workerCount == workers.length)
            return false;
        if (exists(w))
            return false;

        workers[workerCount++] = w;
        return true;
    }

    public double getTotalMonthlyPay() {
        double pay = 0;

        for (int i = 0; i < workerCount; i++)
            pay += workers[i].getMonthlyPay();

        return pay;
    }

```



```java
    public double getAverageHours() {
        double hours = 0;
        int count = 0;

        for (int i = 0; i < workerCount; i++)
            if (workers[i] instanceof HourlyWorker) {
                hours += ((HourlyWorker) workers[i]).getHours();
                count++;
            }

        return count == 0 ? hours : hours / count;
    }

    public String toString() {
        String s = "Business " + name + " (" + workerCount + "):";

        for (int i = 0; i < workerCount; i++)
            s += "\n" + workers[i];

        return s;
    }
}
```



```java
public class BusinessTest {

    public static void main(String [] args) {

        Business b = new Business("SICP");

        b.hireWorker(new IncomeWorker("Saleh", 23, 180000));
        b.hireWorker(new IncomeWorker("Noor", 21, 120000));
        b.hireWorker(new HourlyWorker("Jawaher", 25, 80, 160));
        b.hireWorker(new HourlyWorker("Faisal", 22, 60, 200));

        System.out.println(b);

        System.out.println("Total monthly pay: " + b.getTotalMonthlyPay());

        System.out.println("Average hours: " + b.getAverageHours());

    }
}

```



<terminated> BusinessTest [Java Application] C:\Users\Mahmoud\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.8.v20230831-1047\jre\bin\javaw.exe (20 Feb 2024, 12:26:57 – 12:26:57) [pid: 19360]

```
Business SICP (4):
Worker: Saleh, age: 23, income: 180000.0
Worker: Noor, age: 21, income: 120000.0
Worker: Jawaher, age: 25, rate: 80.0, hours: 160.0
Worker: Faisal, age: 22, rate: 60.0, hours: 200.0
Total monthly pay: 49800.0
Average hours: 180.0
```