Classes

Classes and Objects

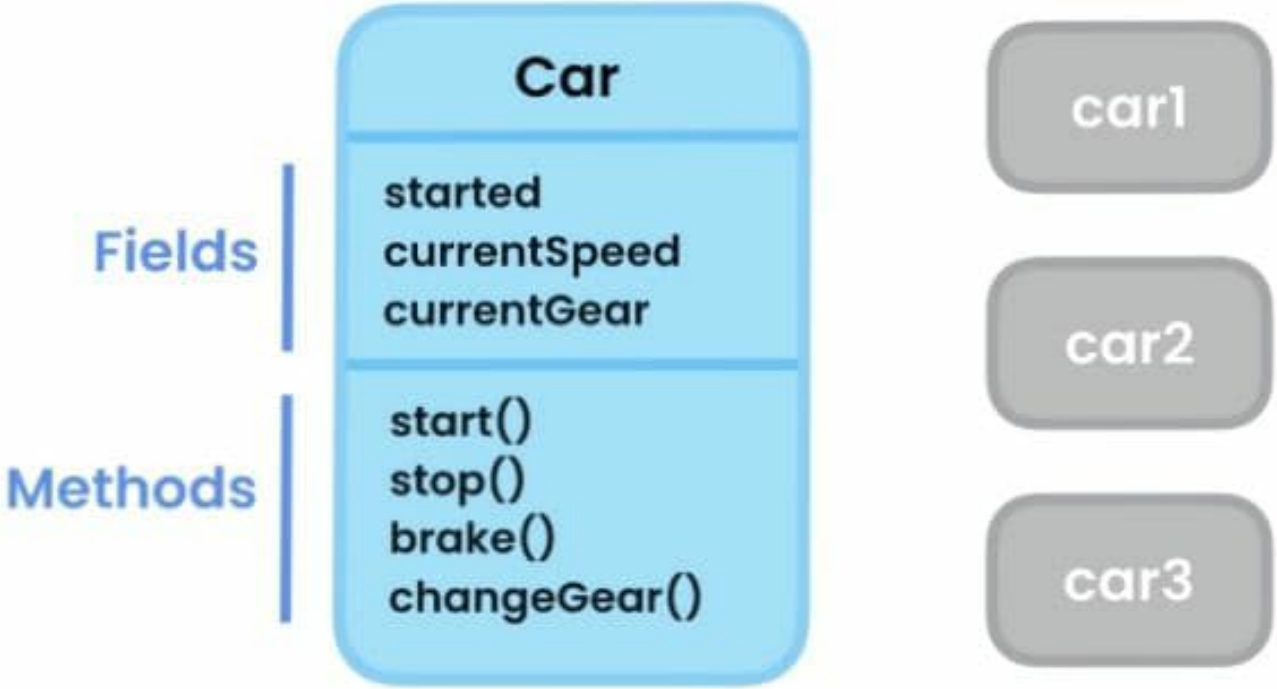
Mosh Hamedani
codewithmosh.com

Class

A blueprint for creating objects

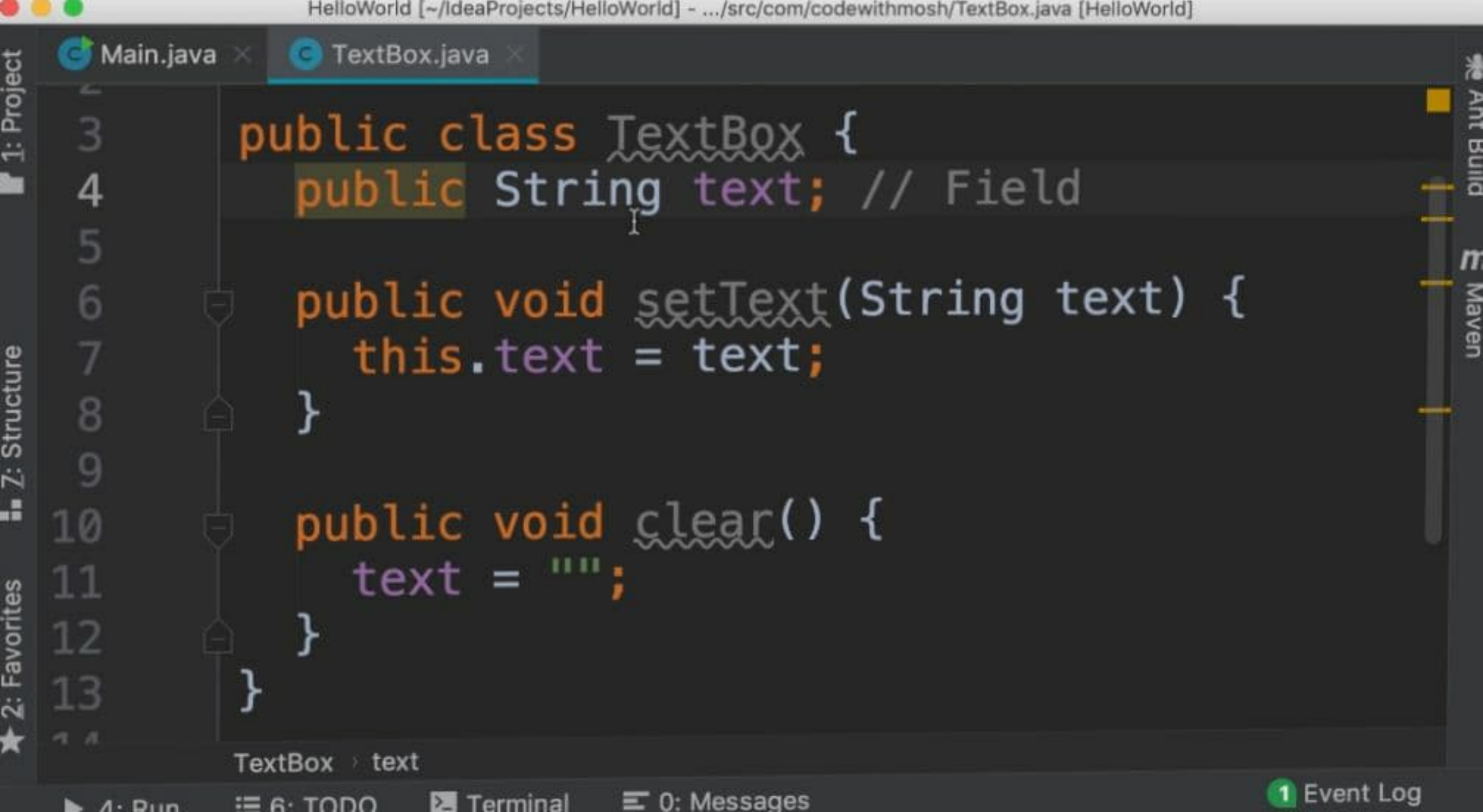
Object

An instance of a class


Car
started
currentSpeed
currentGear
start()
stop()
brake()
changeGear()
Fields
Methods
car1
car2
car3

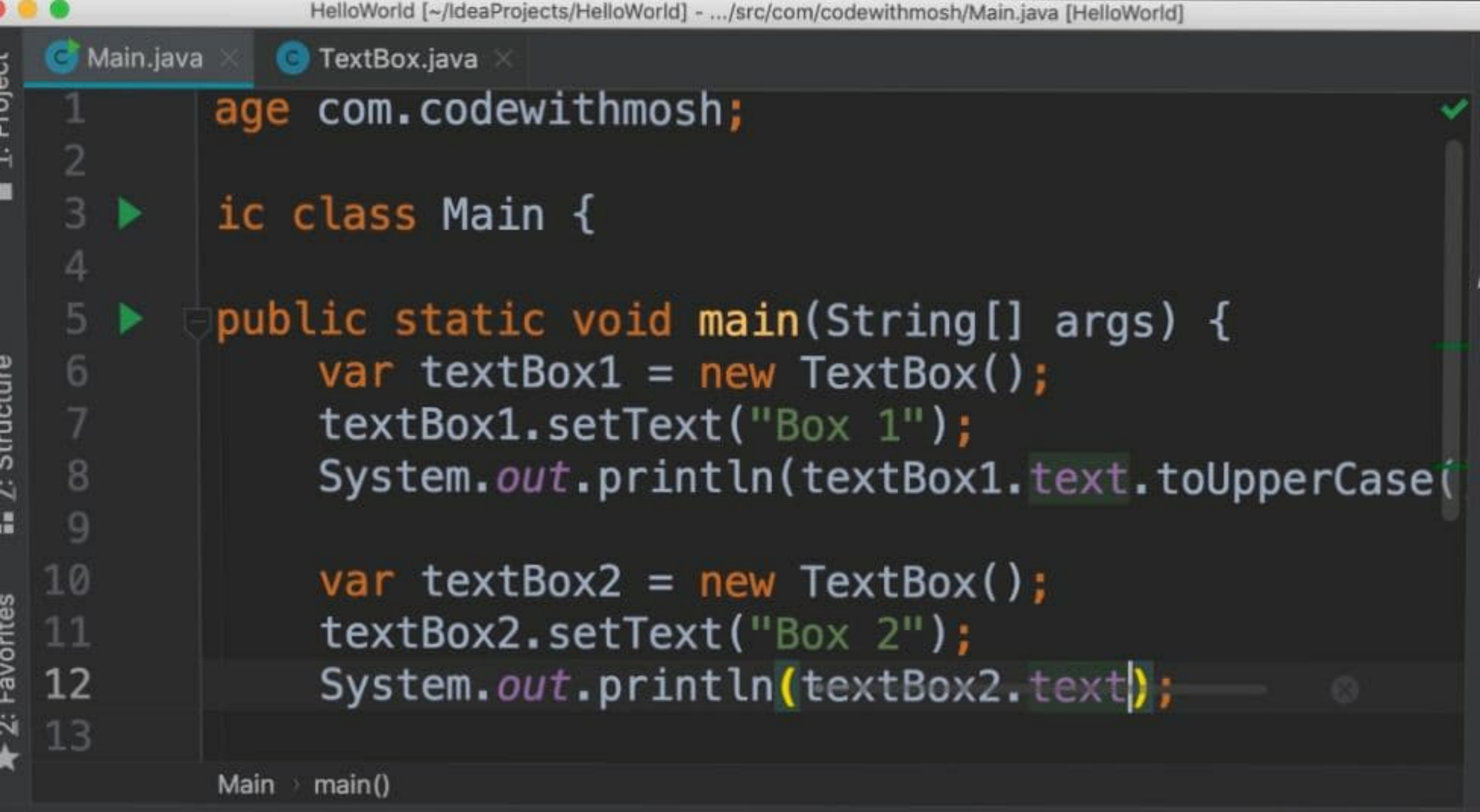
Creating Classes

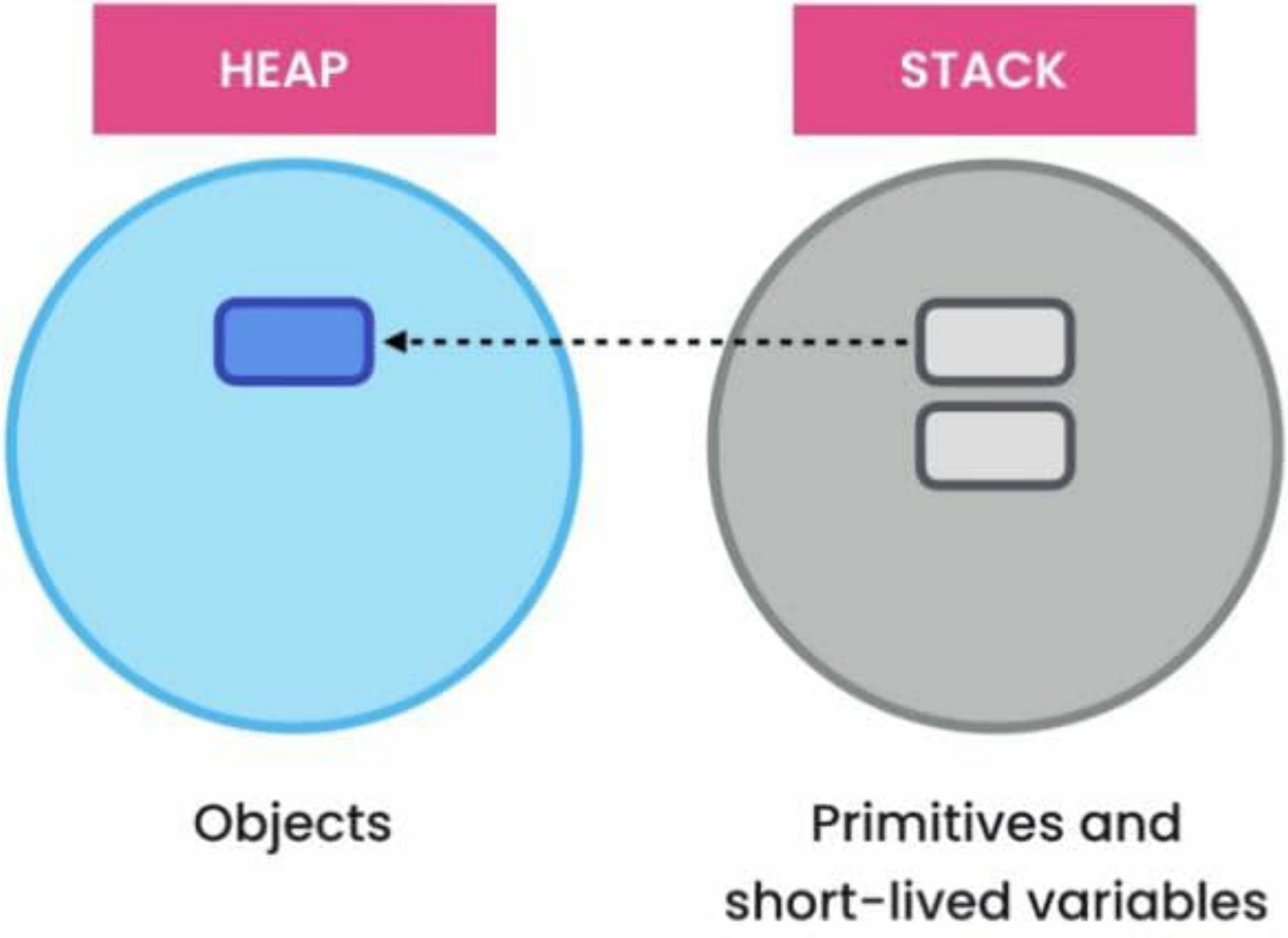
Mosh Hamedani
codewithmosh.com


HelloWorld [~/IdeaProjects/HelloWorld] - .../src/com/codewithmosh/TextBox.java [HelloWorld]
Main.java
TextBox.java
1: Project
2: Favorites
Z: Structure
Ant Build
Maven
3 public class TextBox {
4 public String text; // Field
5
6 public void setText(String text) {
7 this.text = text;
8 }
9
10 public void clear() {
11 text = "";
12 }
13 }
TextBox › text
4: Run
6: TODO
Terminal
0: Messages
1 Event Log

Creating Objects

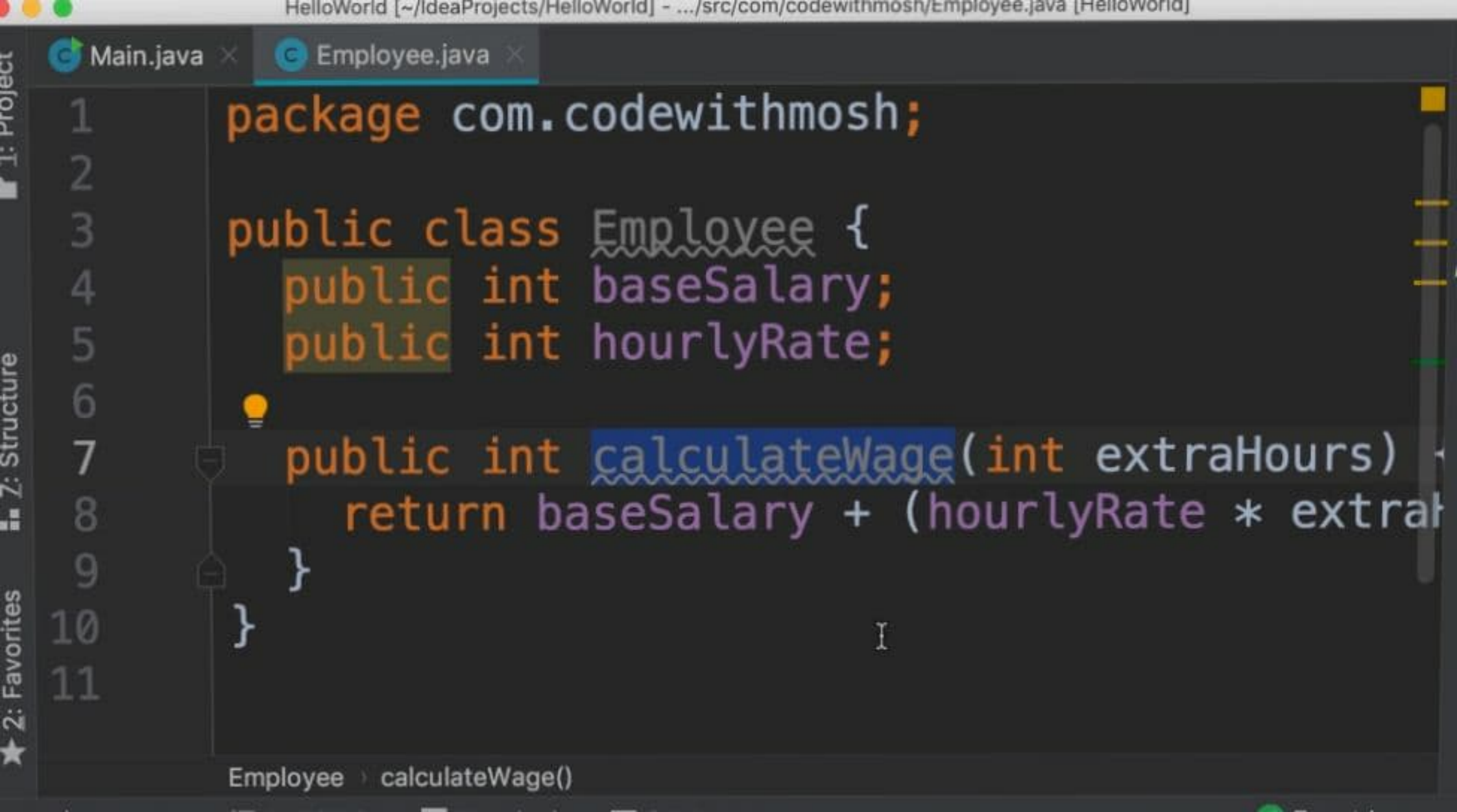
Mosh Hamedani
codewithmosh.com

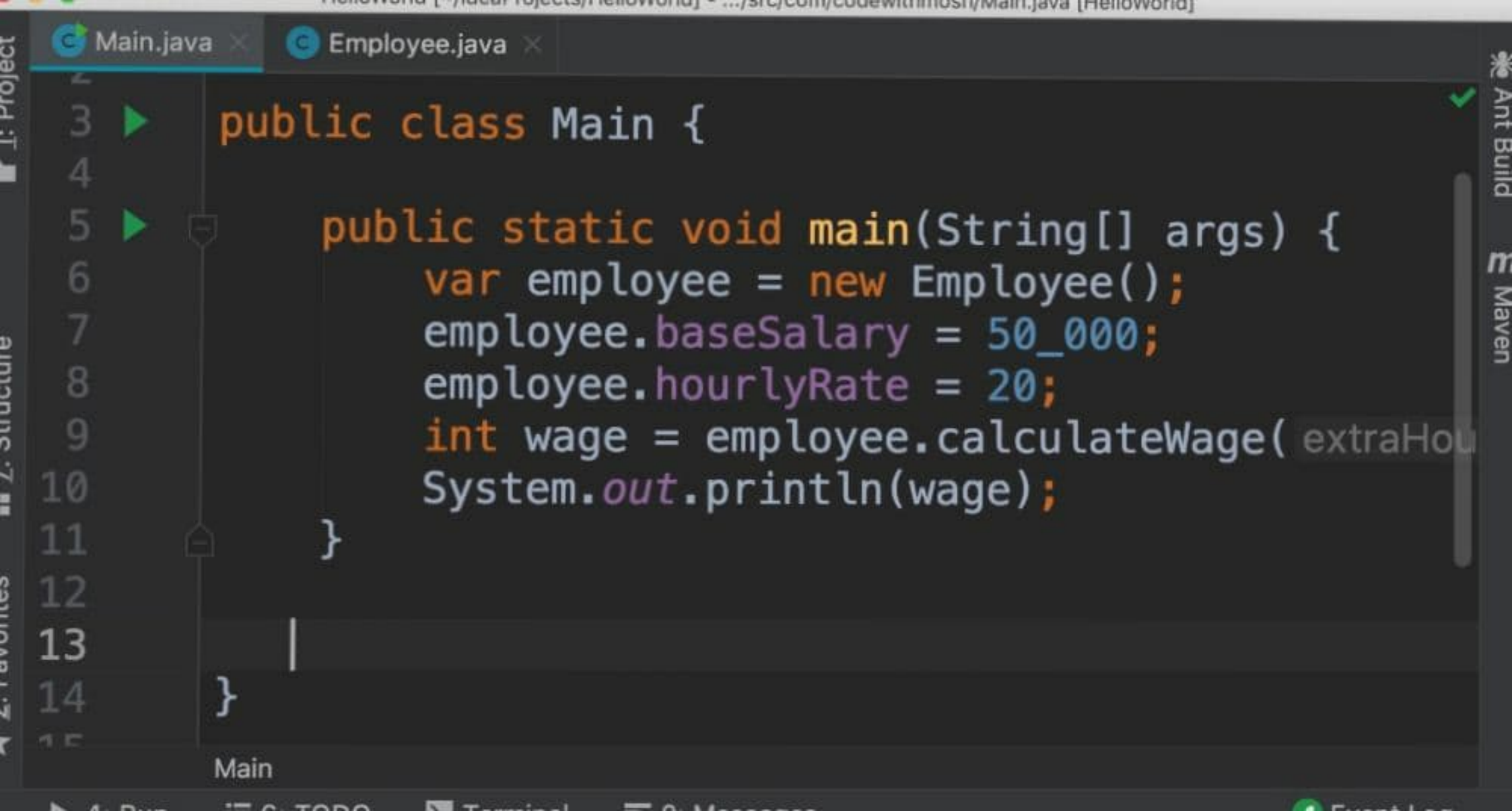

HelloWorld [~/IdeaProjects/HelloWorld] - .../src/com/codewithmosh/Main.java [HelloWorld]
Main.java
TextBox.java
age com.codewithmosh;
ic class Main {
public static void main(String[] args) {
var textBox1 = new TextBox();
textBox1.setText("Box 1");
System.out.println(textBox1.text.toUpperCase(
var textBox2 = new TextBox();
textBox2.setText("Box 2");
System.out.println(textBox2.text);
Main › main()
1: Project
2: Favorites
Z: Structure

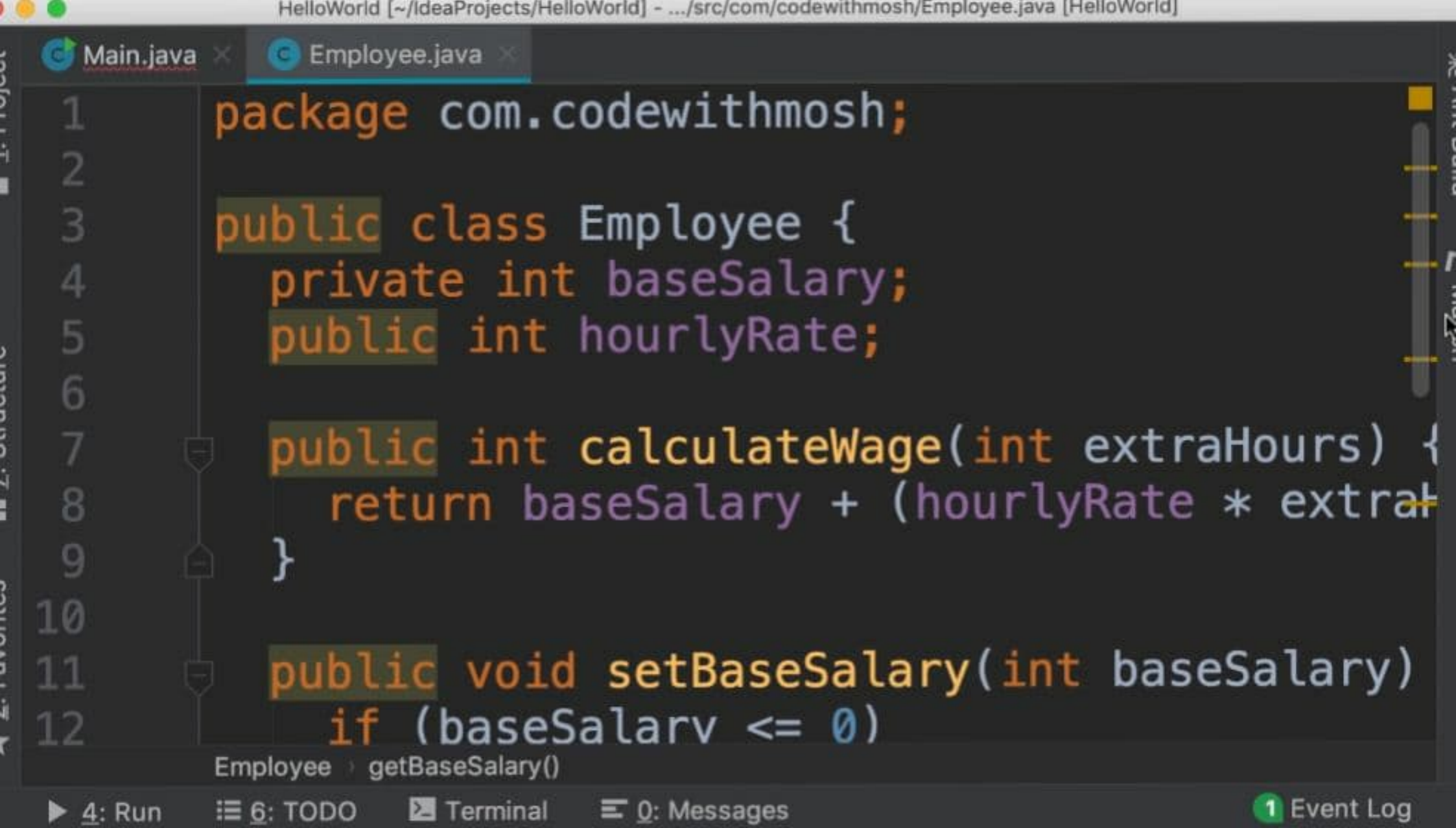

HEAP
STACK
Objects
Primitives and
short-lived variables

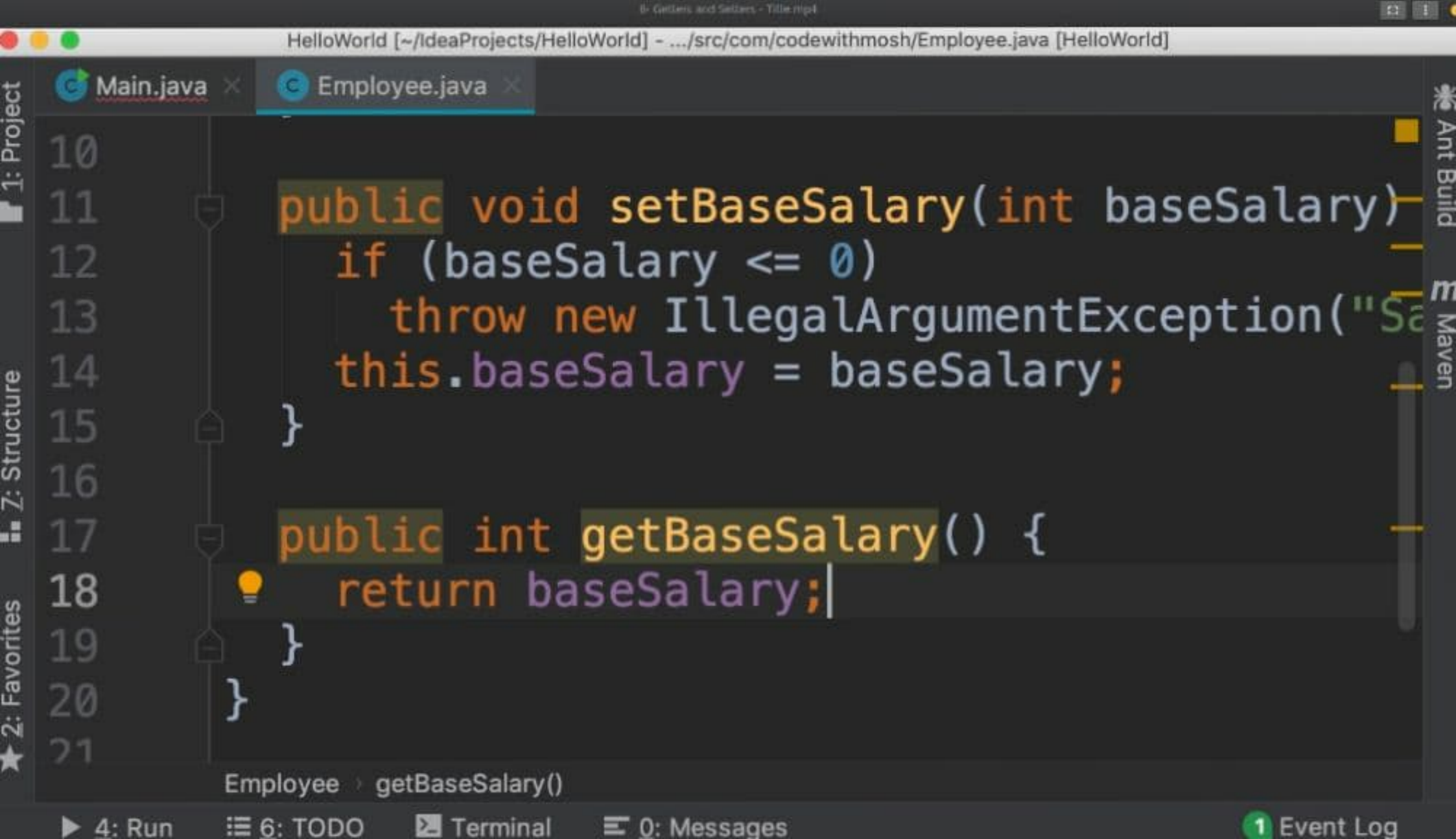
Encapsulation

Mosh Hamedani
codewithmosh.com


HelloWorld [~/IdeaProjects/HelloWorld] - .../src/com/codewithmosh/Employee.java [HelloWorld]
Main.java
Employee.java
1: Project
Z: Structure
2: Favorites
1 package com.codewithmosh;
2
3 public class Employee {
4 public int baseSalary;
5 public int hourlyRate;
6
7 public int calculateWage(int extraHours) {
8 return baseSalary + (hourlyRate * extrah
9 }
10 }
11
Employee › calculateWage()


Main.java
Employee.java
public class Main {
public static void main(String[] args) {
var employee = new Employee();
employee.baseSalary = 50_000;
employee.hourlyRate = 20;
int wage = employee.calculateWage(extraHou
System.out.println(wage);
}
}
Main


HelloWorld [~/IdeaProjects/HelloWorld] - .../src/com/codewithmosh/Employee.java [HelloWorld]
Main.java
Employee.java
1 package com.codewithmosh;
2
3 public class Employee {
4 private int baseSalary;
5 public int hourlyRate;
6
7 public int calculateWage(int extraHours) {
8 return baseSalary + (hourlyRate * extraH
9 }
10
11 public void setBaseSalary(int baseSalary)
12 if (baseSalary <= 0)
Employee › getBaseSalary()
4: Run
6: TODO
Terminal
0: Messages
1 Event Log


HelloWorld [~/IdeaProjects/HelloWorld] - .../src/com/codewithmosh/Employee.java [HelloWorld]
Main.java
Employee.java
1: Project
Z: Structure
2: Favorites
Ant Build
Maven
10
11 public void setBaseSalary(int baseSalary)
12 if (baseSalary <= 0)
13 throw new IllegalArgumentException("Sa
14 this.baseSalary = baseSalary;
15 }
16
17 public int getBaseSalary() {
18 return baseSalary;
19 }
20 }
21
Employee › getBaseSalary()
4: Run
6: TODO
Terminal
0: Messages
1 Event Log

Abstraction

Mosh Hamedani
codewithmosh.com

Reduce complexity by
hiding unnecessary details

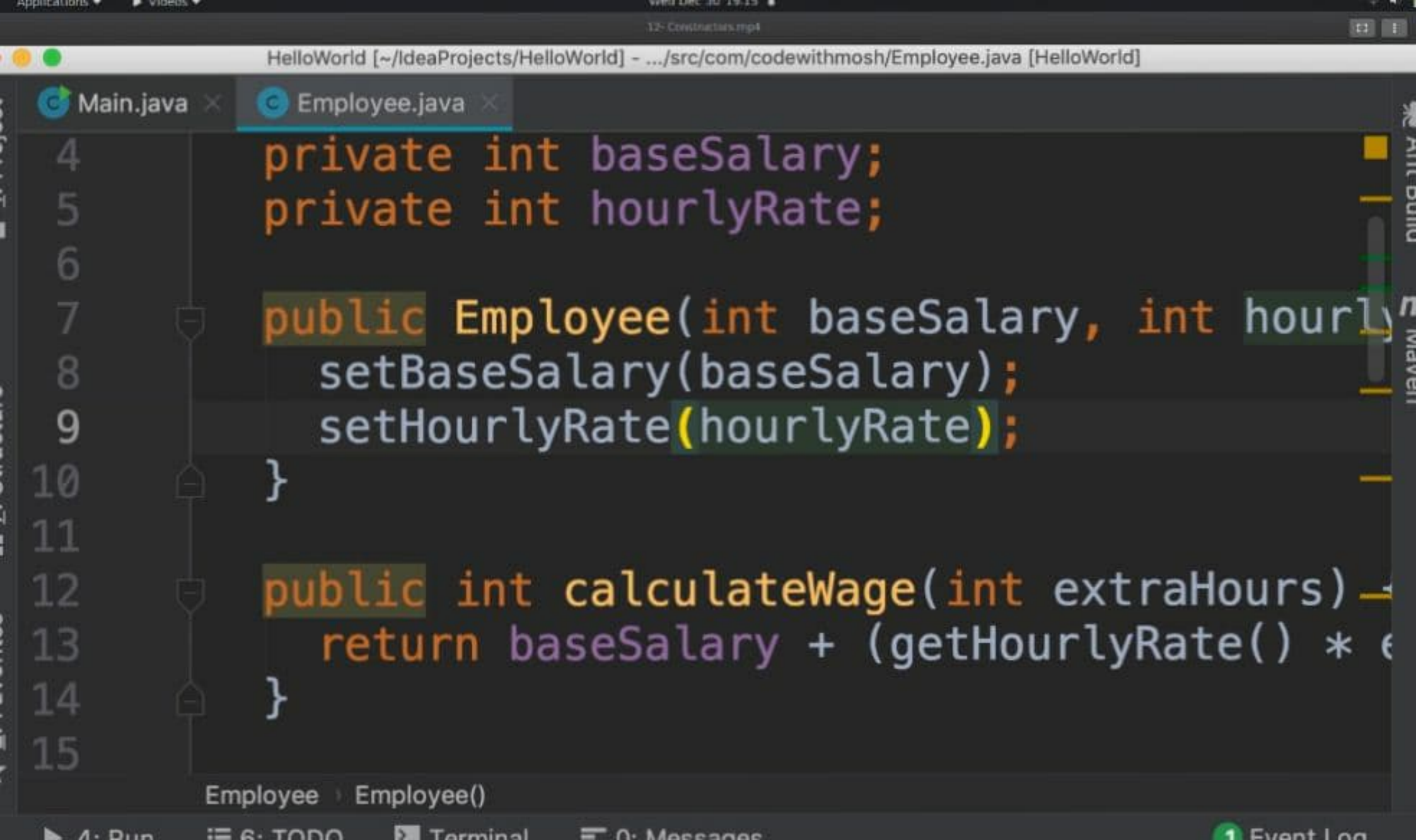
Coupling

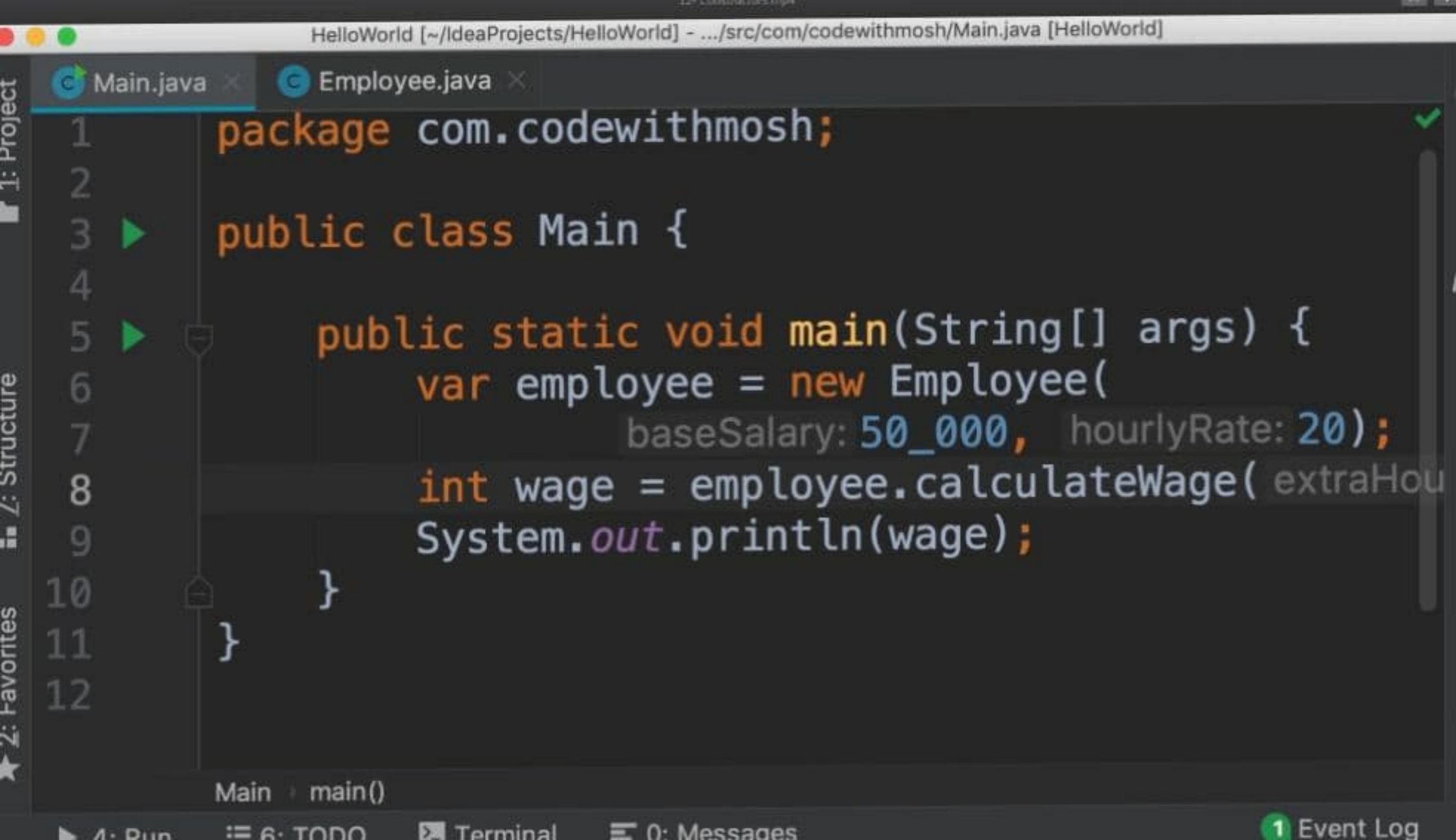
Mosh Hamedani
codewithmosh.com

The level of dependency between classes

Constructors

Mosh Hamedani
codewithmosh.com


HelloWorld [~/IdeaProjects/HelloWorld] - .../src/com/codewithmosh/Employee.java [HelloWorld]
Main.java
Employee.java
private int baseSalary;
private int hourlyRate;
public Employee(int baseSalary, int hourly
setBaseSalary(baseSalary);
setHourlyRate(hourlyRate);
}
public int calculateWage(int extraHours)
return baseSalary + (getHourlyRate() * e
}
Employee › Employee()
4: Run
6: TODO
Terminal
0: Messages
Event Log


HelloWorld [~/IdeaProjects/HelloWorld] - .../src/com/codewithmosh/Main.java [HelloWorld]
Main.java
Employee.java
1: Project
Z: Structure
2: Favorites
package com.codewithmosh;
public class Main {
public static void main(String[] args) {
var employee = new Employee(
baseSalary: 50_000, hourlyRate: 20);
int wage = employee.calculateWage(extraHou
System.out.println(wage);
}
}
Main › main()
4: Run
6: TODO
Terminal
0: Messages
1 Event Log

Constructor Overloading

Mosh Hamedani
codewithmosh.com

Main.java | Employee.java

```
5       private int hourlyRate;
6
7       public Employee(int baseSalary) {
8           setBaseSalary(baseSalary);
9           setHourlyRate(0);
10      }
11
12      public Employee(int baseSalary, int hourlyRate) {
13          setBaseSalary(baseSalary);
14          setHourlyRate(hourlyRate);
15      }
16
17      public int calculateWage(int extraHours) {
18          return baseSalary + (getHourlyRate() * extraHours);
19      }
```

Employee › Employee()

Main.java | Employee.java

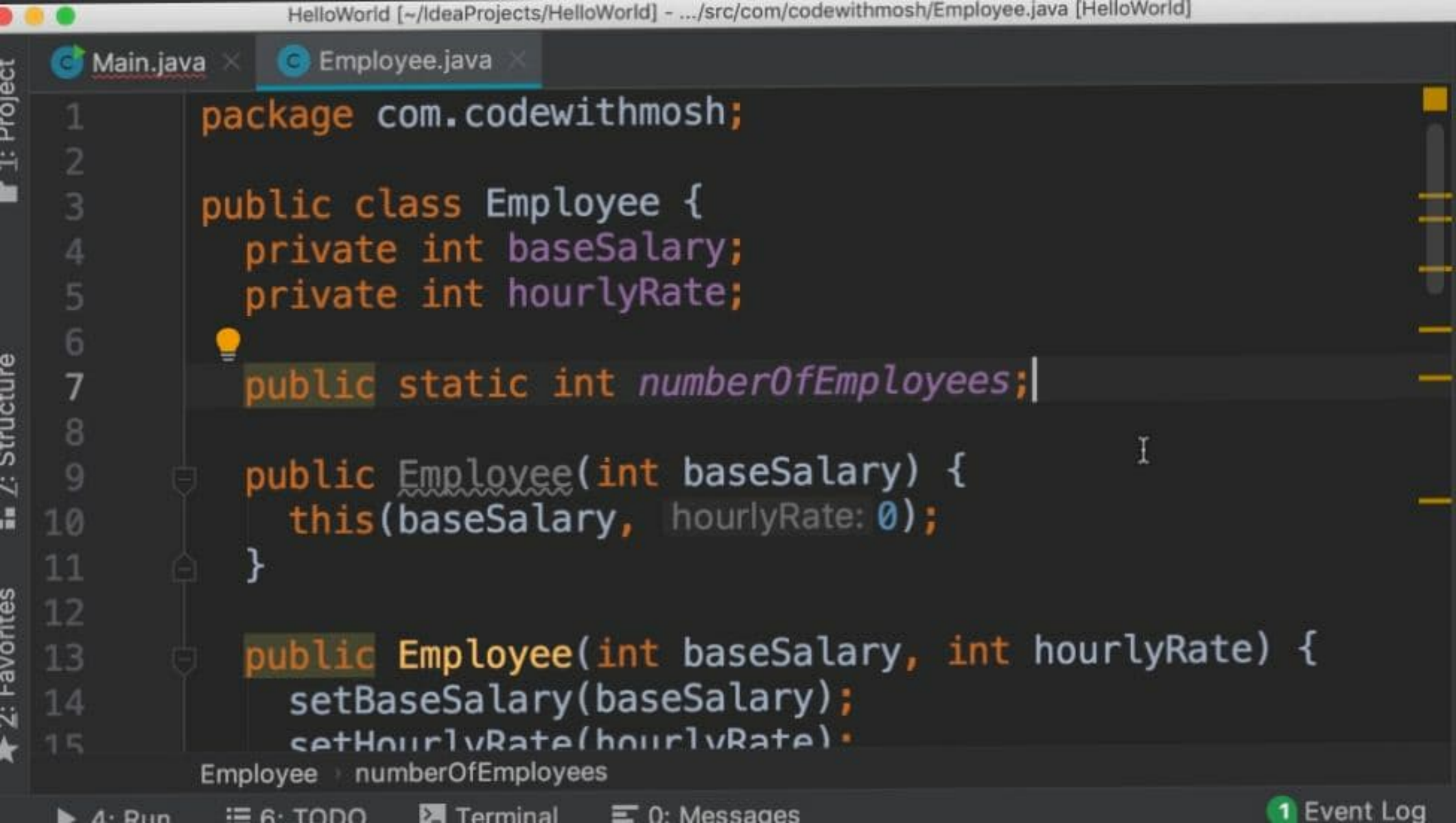
```
6
7       public Employee(int baseSalary) {
8           this(baseSalary,  hourlyRate: 0);
9       }
10
11      public Employee(int baseSalary, int hourlyRate) {
12          setBaseSalary(baseSalary);
13          setHourlyRate(hourlyRate);
14      }
15
16      public int calculateWage(int extraHours) {
17          return baseSalary + (getHourlyRate() * extraHours);
18      }
19
20      public int calculateWage() {
```

Employee › Employee()

4: Run | 6: TODO | Terminal | 0: Messages | 1 Event Log

Static Members

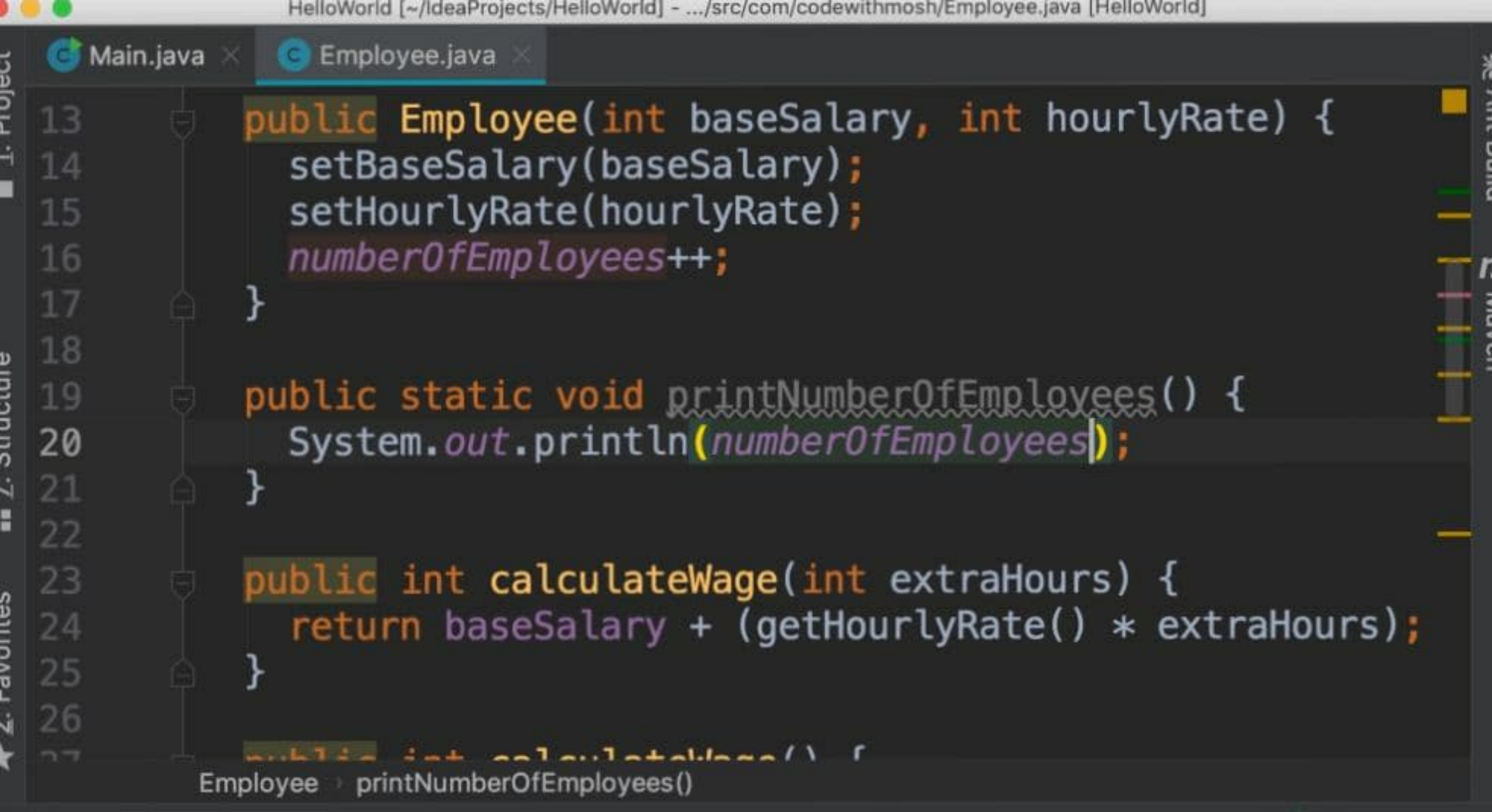
Mosh Hamedani
codewithmosh.com

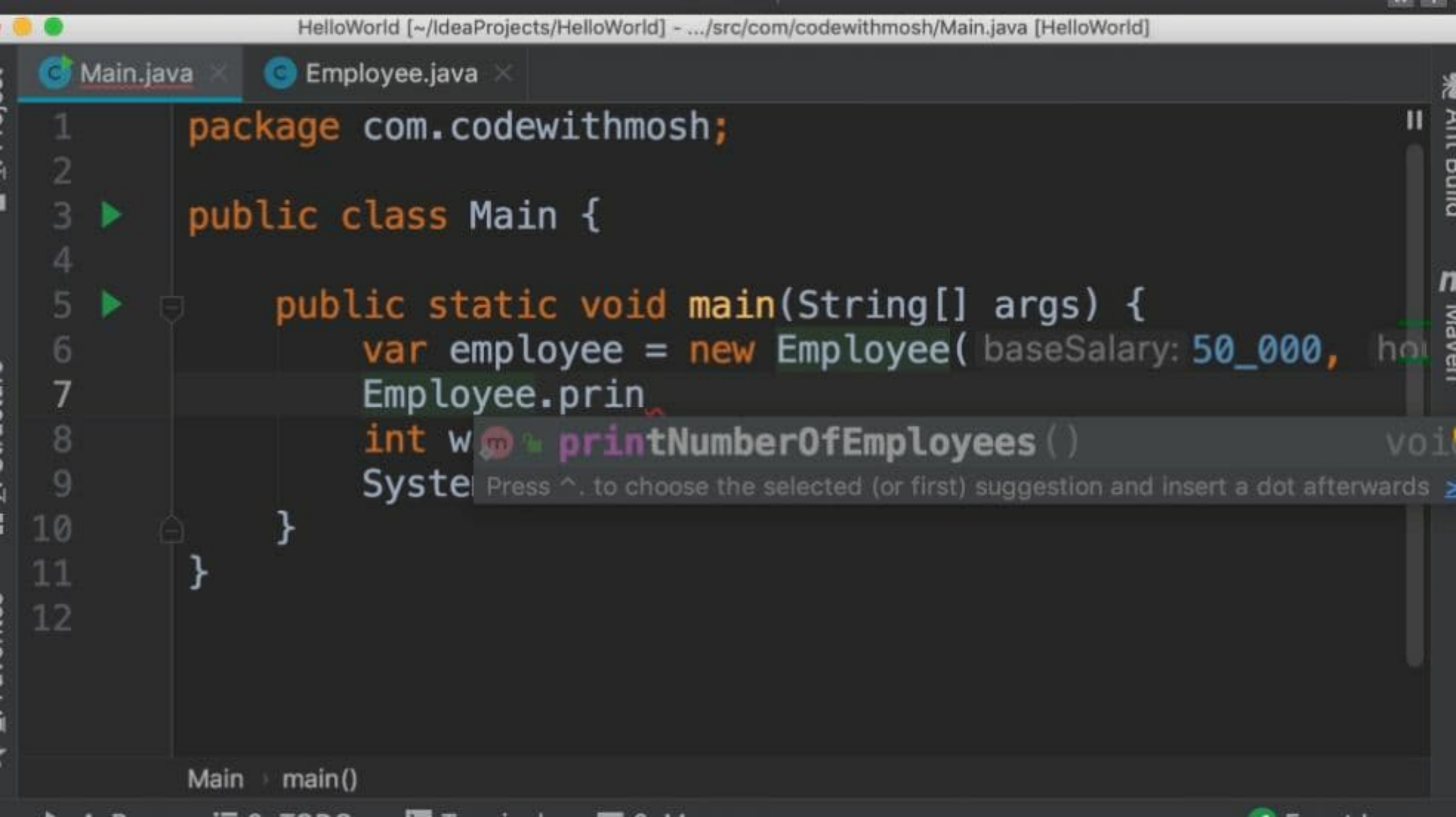

HelloWorld [~/IdeaProjects/HelloWorld] - .../src/com/codewithmosh/Employee.java [HelloWorld]
Main.java
Employee.java
1: Project
2: Favorites
Z: Structure
1 package com.codewithmosh;
2
3 public class Employee {
4 private int baseSalary;
5 private int hourlyRate;
6
7 public static int numberOfEmployees;
8
9 public Employee(int baseSalary) {
10 this(baseSalary, hourlyRate: 0);
11 }
12
13 public Employee(int baseSalary, int hourlyRate) {
14 setBaseSalary(baseSalary);
15 setHourlyRate(hourlyRate);
Employee › numberOfEmployees
4: Run
6: TODO
Terminal
0: Messages
1 Event Log

Main.java | Employee.java

```
7   public static int numberOfEmployees;
8
9   public Employee(int baseSalary) {
10    this(baseSalary, hourlyRate: 0);
11  }
12
13  public Employee(int baseSalary, int hourlyRate) {
14    setBaseSalary(baseSalary);
15    setHourlyRate(hourlyRate);
16    numberOfEmployees++;
17  }
18
19  public int calculateWage(int extraHours) {
20    return baseSalary + (getHourlyRate() * extraHours);
21  }
```

Employee › Employee()


HelloWorld [~/IdeaProjects/HelloWorld] - .../src/com/codewithmosh/Employee.java [HelloWorld]
Main.java
Employee.java
1. Project
2. Structure
2. Favorites
Ant Build
13 public Employee(int baseSalary, int hourlyRate) {
14 setBaseSalary(baseSalary);
15 setHourlyRate(hourlyRate);
16 numberOfEmployees++;
17 }
18
19 public static void printNumberOfEmployees() {
20 System.out.println(numberOfEmployees);
21 }
22
23 public int calculateWage(int extraHours) {
24 return baseSalary + (getHourlyRate() * extraHours);
25 }
26
Employee › printNumberOfEmployees()


HelloWorld [~/IdeaProjects/HelloWorld] - .../src/com/codewithmosh/Main.java [HelloWorld]
Main.java
Employee.java
1 package com.codewithmosh;
2
3 public class Main {
4
5 public static void main(String[] args) {
6 var employee = new Employee(baseSalary: 50_000,
7 Employee.prin
8 int w
9 Syste
10 }
11 }
12
printNumberOfEmployees()
Press ^. to choose the selected (or first) suggestion and insert a dot afterwards
Main › main()