



Java Assignment 3

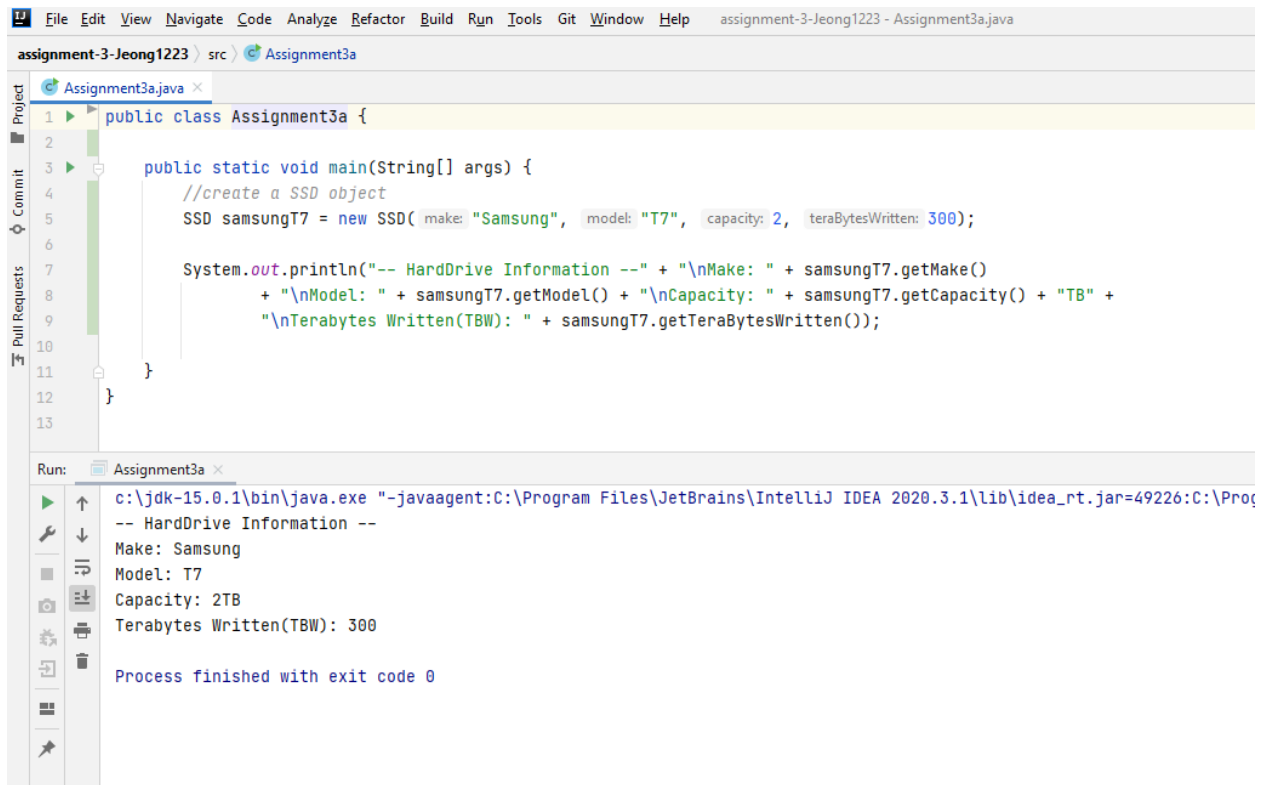
Jeong Eun Jang (W0451032)

PROG 1400-700

Brian Shewan

April 2nd, 2021

1. Assignment3a



The screenshot displays the IntelliJ IDEA interface. The top menu bar includes File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, Git, Window, and Help. The project name is 'assignment-3-Jeong1223' and the current file is 'Assignment3a.java'. The code editor shows the following Java code:

```
1 public class Assignment3a {  
2  
3     public static void main(String[] args) {  
4         //create a SSD object  
5         SSD samsungT7 = new SSD( make: "Samsung", model: "T7", capacity: 2, teraBytesWritten: 300);  
6  
7         System.out.println("-- HardDrive Information --" + "\nMake: " + samsungT7.getMake()  
8             + "\nModel: " + samsungT7.getModel() + "\nCapacity: " + samsungT7.getCapacity() + "TB" +  
9             "\nTerabytes Written(TBW): " + samsungT7.getTeraBytesWritten());  
10  
11     }  
12 }  
13
```

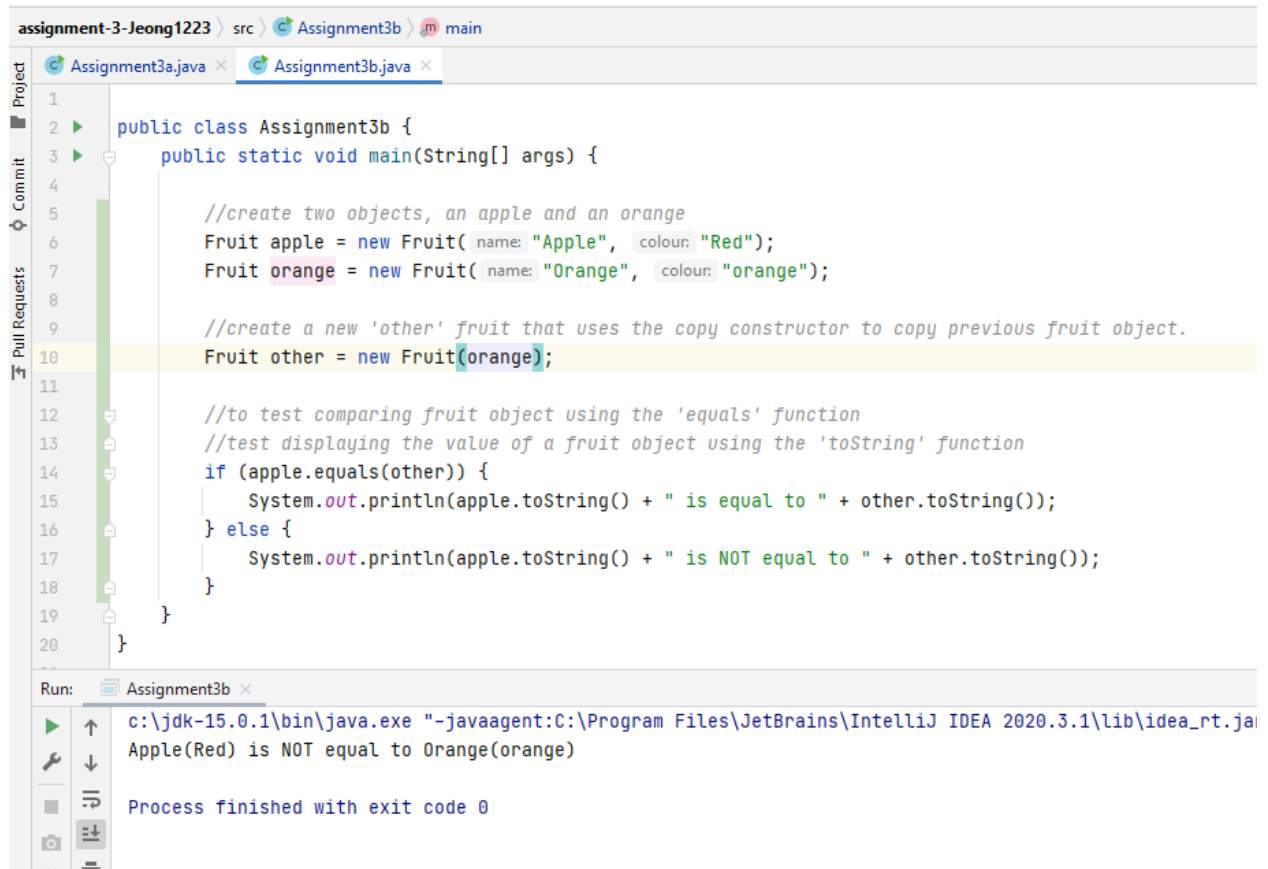
The Run window at the bottom shows the execution of the program. The command used is:

```
c:\jdk-15.0.1\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2020.3.1\lib\idea_rt.jar=49226:C:\Progr
```

The output of the program is:

```
-- HardDrive Information --  
Make: Samsung  
Model: T7  
Capacity: 2TB  
Terabytes Written(TBW): 300  
  
Process finished with exit code 0
```

2. Assignment3b



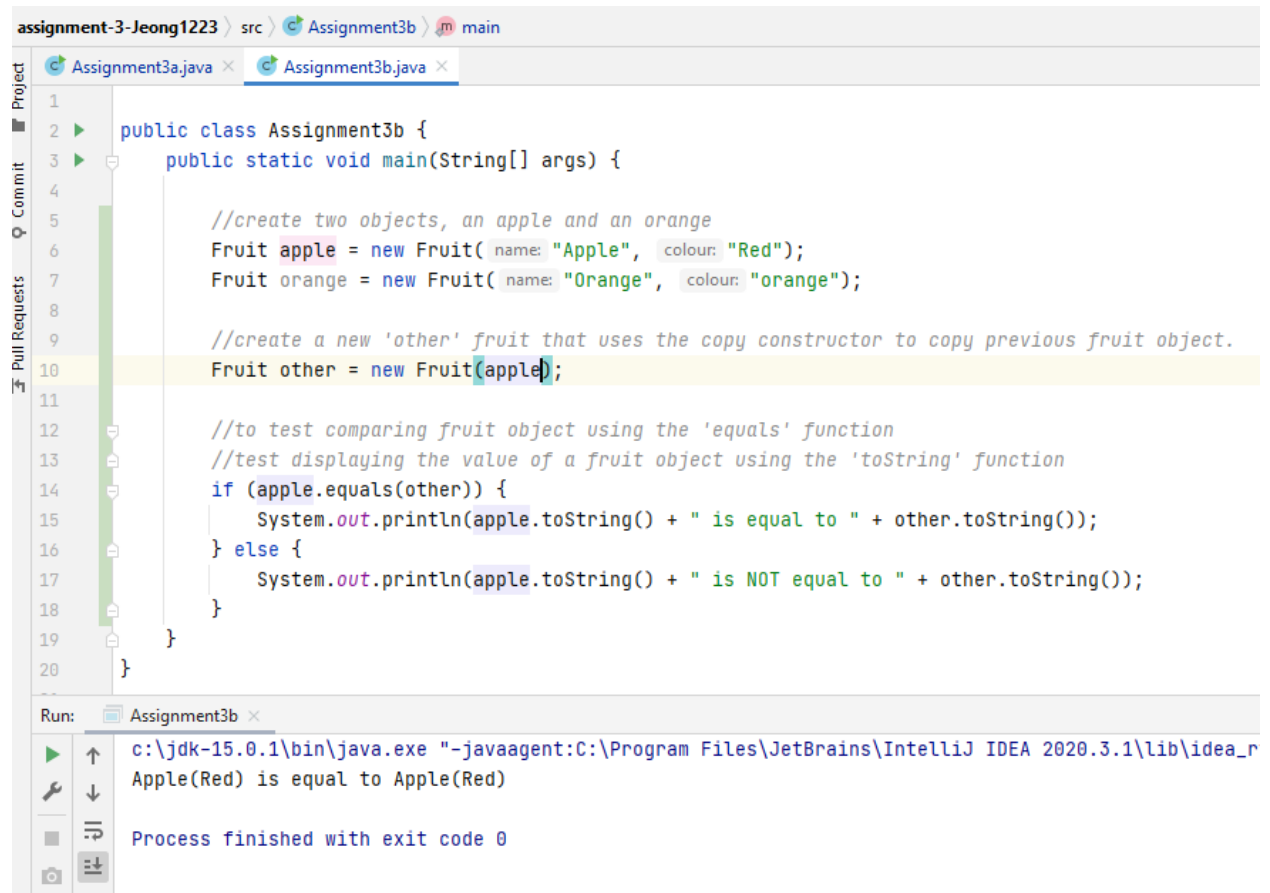
The screenshot displays the IntelliJ IDEA IDE interface. The top toolbar includes icons for Project, Commit, and Pull Requests. The main editor window shows the file `Assignment3b.java` with the following Java code:

```
1 public class Assignment3b {
2     public static void main(String[] args) {
3
4         //create two objects, an apple and an orange
5         Fruit apple = new Fruit( name: "Apple", colour: "Red");
6         Fruit orange = new Fruit( name: "Orange", colour: "orange");
7
8         //create a new 'other' fruit that uses the copy constructor to copy previous fruit object.
9         Fruit other = new Fruit(orange);
10
11         //to test comparing fruit object using the 'equals' function
12         //test displaying the value of a fruit object using the 'toString' function
13         if (apple.equals(other)) {
14             System.out.println(apple.toString() + " is equal to " + other.toString());
15         } else {
16             System.out.println(apple.toString() + " is NOT equal to " + other.toString());
17         }
18     }
19 }
20 }
```

Below the code editor, the 'Run' tab is active, showing the execution command and output:

```
Run: Assignment3b
c:\jdk-15.0.1\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2020.3.1\lib\idea_rt.jar
Apple(Red) is NOT equal to Orange(orange)

Process finished with exit code 0
```

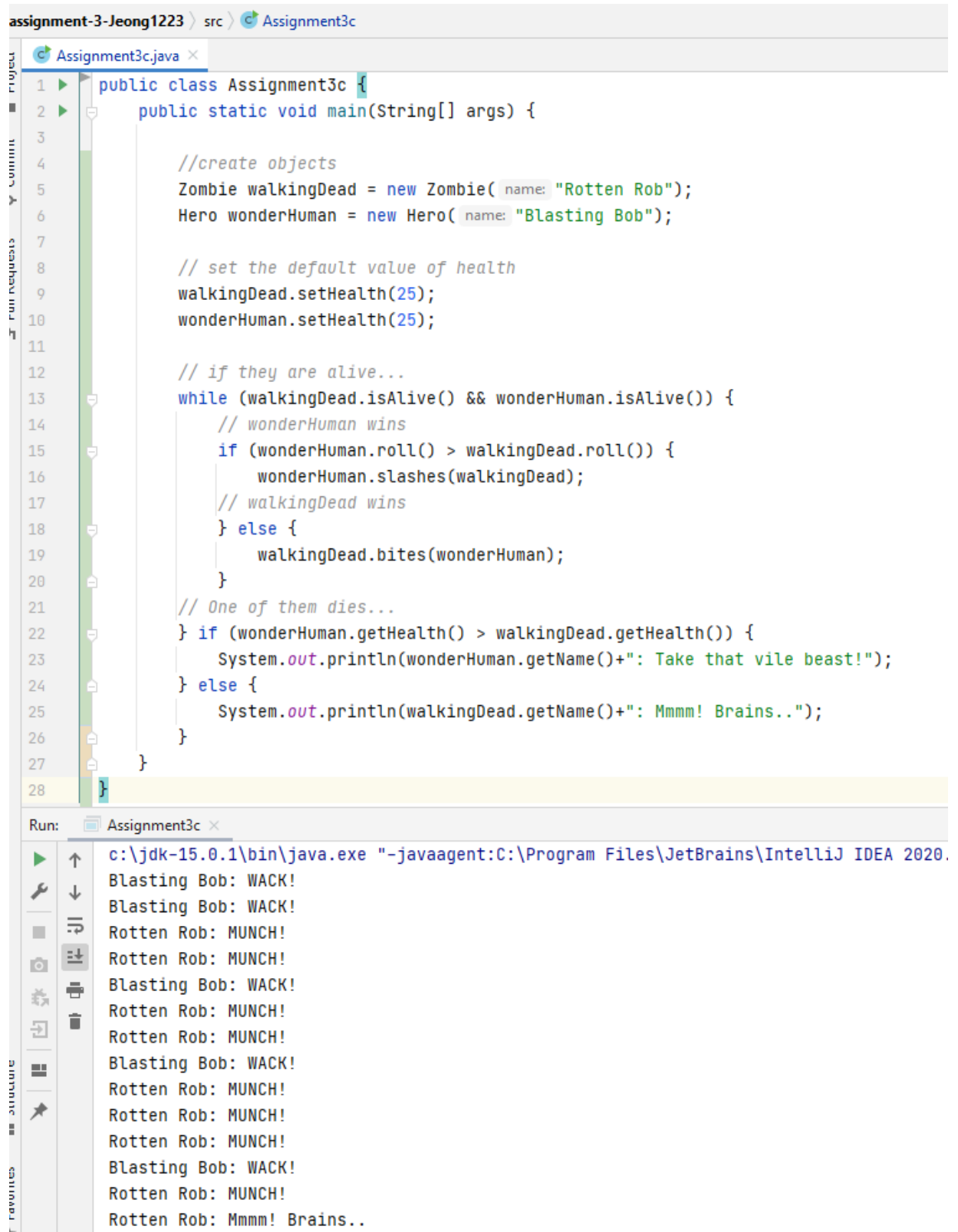


The screenshot displays the IntelliJ IDEA IDE interface. At the top, the project path is 'assignment-3-Jeong1223' with subdirectories 'src' and 'Assignment3b'. The main editor window shows 'Assignment3b.java' with the following code:

```
1 public class Assignment3b {
2
3     public static void main(String[] args) {
4
5         //create two objects, an apple and an orange
6         Fruit apple = new Fruit( name: "Apple", colour: "Red");
7         Fruit orange = new Fruit( name: "Orange", colour: "orange");
8
9         //create a new 'other' fruit that uses the copy constructor to copy previous fruit object.
10        Fruit other = new Fruit(apple);
11
12        //to test comparing fruit object using the 'equals' function
13        //test displaying the value of a fruit object using the 'toString' function
14        if (apple.equals(other)) {
15            System.out.println(apple.toString() + " is equal to " + other.toString());
16        } else {
17            System.out.println(apple.toString() + " is NOT equal to " + other.toString());
18        }
19    }
20 }
```

Below the code editor, the 'Run' tab is active, showing the command executed: 'c:\jdk-15.0.1\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2020.3.1\lib\idea_r' and the output: 'Apple(Red) is equal to Apple(Red)'. The process finished with exit code 0.

3. Assignment3c

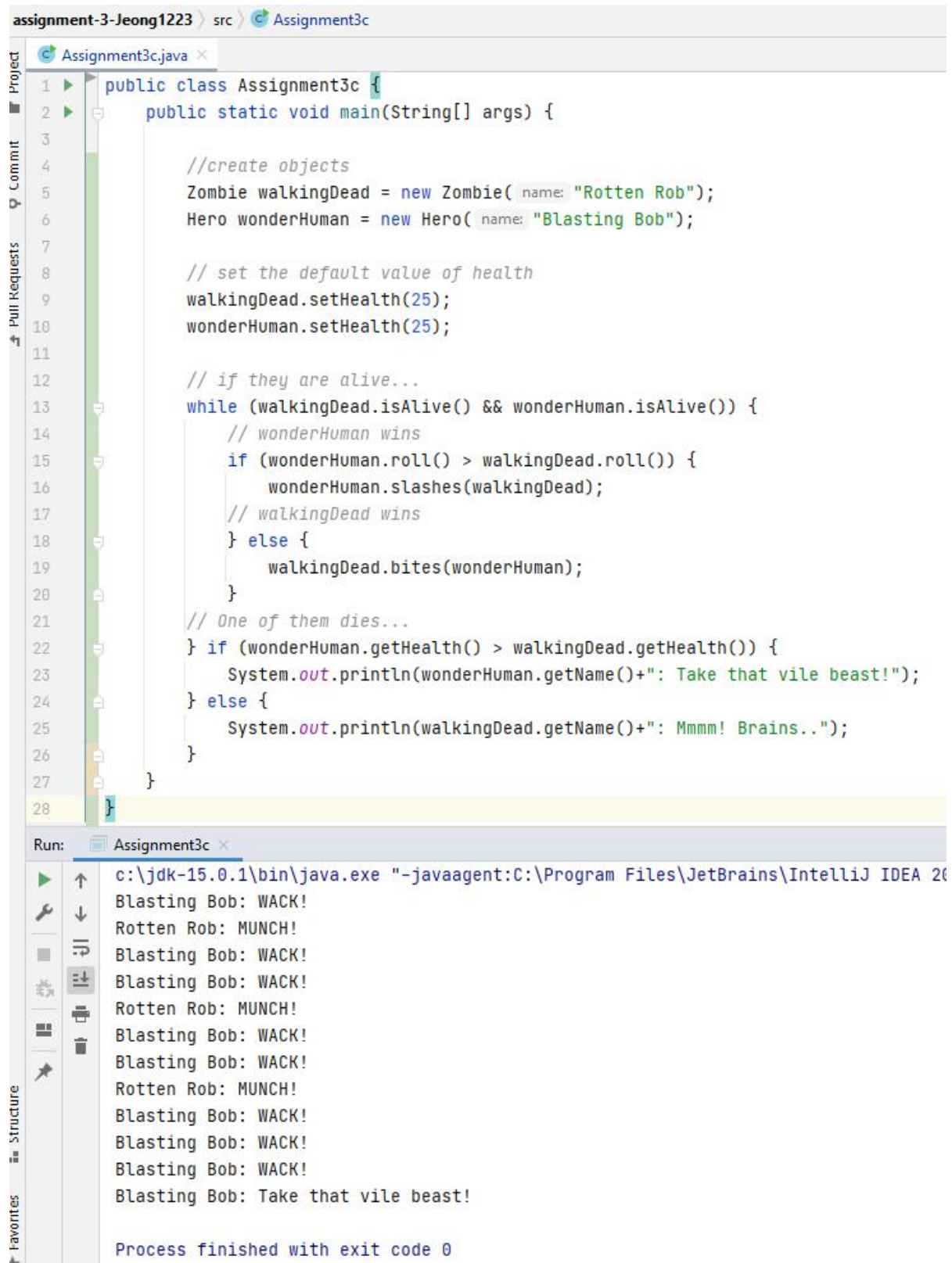


The screenshot displays an IDE window titled "assignment-3-Jeong1223" with a sub-window "Assignment3c". The code editor shows the following Java code:

```
1 public class Assignment3c {
2     public static void main(String[] args) {
3
4         //create objects
5         Zombie walkingDead = new Zombie( name: "Rotten Rob");
6         Hero wonderHuman = new Hero( name: "Blasting Bob");
7
8         // set the default value of health
9         walkingDead.setHealth(25);
10        wonderHuman.setHealth(25);
11
12        // if they are alive...
13        while (walkingDead.isAlive() && wonderHuman.isAlive()) {
14            // wonderHuman wins
15            if (wonderHuman.roll() > walkingDead.roll()) {
16                wonderHuman.slashes(walkingDead);
17                // walkingDead wins
18            } else {
19                walkingDead.bites(wonderHuman);
20            }
21            // One of them dies...
22        } if (wonderHuman.getHealth() > walkingDead.getHealth()) {
23            System.out.println(wonderHuman.getName()+": Take that vile beast!");
24        } else {
25            System.out.println(walkingDead.getName()+": Mmmm! Brains..");
26        }
27    }
28 }
```

The "Run" tab at the bottom shows the execution output for "Assignment3c":

```
c:\jdk-15.0.1\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2020.
Blasting Bob: WACK!
Blasting Bob: WACK!
Rotten Rob: MUNCH!
Rotten Rob: MUNCH!
Blasting Bob: WACK!
Rotten Rob: MUNCH!
Rotten Rob: MUNCH!
Blasting Bob: WACK!
Rotten Rob: MUNCH!
Rotten Rob: MUNCH!
Blasting Bob: WACK!
Rotten Rob: MUNCH!
Rotten Rob: MUNCH!
Blasting Bob: WACK!
Rotten Rob: MUNCH!
Rotten Rob: Mmmm! Brains..
```



The screenshot displays the IntelliJ IDEA interface. The top pane shows the source code for `Assignment3c.java`. The code defines a `public class Assignment3c` with a `main` method. It creates two objects: `Zombie walkingDead` with name "Rotten Rob" and `Hero wonderHuman` with name "Blasting Bob". Both are initialized with a health of 25. A `while` loop runs as long as both are alive. Inside, it uses `roll()` to determine the winner. If `wonderHuman` wins, it calls `slashes(walkingDead)`; if `walkingDead` wins, it calls `bites(wonderHuman)`. The loop ends when one dies, and the winner's name and action are printed. The bottom pane shows the program's execution output, which repeats the actions of "Blasting Bob" (WACK!) and "Rotten Rob" (MUNCH!) several times before "Blasting Bob" finally wins and prints "Take that vile beast!". The process ends with exit code 0.

```
1 public class Assignment3c {
2     public static void main(String[] args) {
3
4         //create objects
5         Zombie walkingDead = new Zombie( name: "Rotten Rob");
6         Hero wonderHuman = new Hero( name: "Blasting Bob");
7
8         // set the default value of health
9         walkingDead.setHealth(25);
10        wonderHuman.setHealth(25);
11
12        // if they are alive...
13        while (walkingDead.isAlive() && wonderHuman.isAlive()) {
14            // wonderHuman wins
15            if (wonderHuman.roll() > walkingDead.roll()) {
16                wonderHuman.slashes(walkingDead);
17            // walkingDead wins
18            } else {
19                walkingDead.bites(wonderHuman);
20            }
21            // One of them dies...
22        } if (wonderHuman.getHealth() > walkingDead.getHealth()) {
23            System.out.println(wonderHuman.getName()+" Take that vile beast!");
24        } else {
25            System.out.println(walkingDead.getName()+" Mmmm! Brains..");
26        }
27    }
28 }
```

Run: Assignment3c

c:\jdk-15.0.1\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2020.3\lib\idea_rt.jar=12737:C:\Program Files\JetBrains\IntelliJ IDEA 2020.3\bin" -Dfile.encoding=UTF-8

Blasting Bob: WACK!
Rotten Rob: MUNCH!
Blasting Bob: WACK!
Blasting Bob: WACK!
Rotten Rob: MUNCH!
Blasting Bob: WACK!
Blasting Bob: WACK!
Rotten Rob: MUNCH!
Blasting Bob: WACK!
Blasting Bob: WACK!
Blasting Bob: WACK!
Blasting Bob: Take that vile beast!

Process finished with exit code 0