

Pg. 129 – 130, Java Programming *A comprehensive Introduction*

Class and Object expanded - Continued

Section 1: Define / Answer

“programmer created” class- How do “programmer created methods” using **void**, differ from using created using **return**? Briefly - Define how return works.

The differences between using void and using return are if you are using void, you couldn't return any interger type such as Double, float, Int. But If you are using return, you must return the interger type.

And if you use void, the systax should be

```
void something() {  
  
}
```

and if you use return, the systax should be

```
public int something() {  
  
int l = 0  
  
return l;  
  
}
```

Pg. 123, Java Programming *A comprehensive Introduction*

Programming Assignment

Task 1- Create a computer program that will calculate the range for 3 different vehicles.

The program should create a “programmer created” class, where 3 **int objects** are created passengers, fuel capacity, mpg.

Set-up the program so the user can manually input the values for passengers, fuel capacity, mpg for the 3 created vehicles.

Create a **return** method inside the “programmer created” class to calculate vehicle range.

range = fuel capacity * miles per gallon.

Each Vehicle type should have unique values for number of passengers, fuel capacity, and miles per gallon.

Follow the sample below and return information on 3 vehicle types.

Sample Output: // Create similar output for 3 Vehicle Types

The minivan carries= 7

The minivan has a fuel capacity of = 16

The minivan mpg = 21

The minivan has a range of: 336 miles

```
1 package javaapplication1;
2
3
4 import java.util.Scanner;
5
6 class Vehicle {
7
8     int passengers = 0;
9     int fuelcapacity = 0;
10    int mpg = 0;
11
12    public int range(int newValue, int newValue2) {
13
14        fuelcapacity = newValue;
15        mpg = newValue2;
16
17        return fuelcapacity * mpg;
18    }
19
20 }
21
22 public class JavaApplication1 {
23
24    public static void main(String[] args) {
25
26        Scanner input = new Scanner(System.in);
27
28        //minivan
29        Vehicle minivan = new Vehicle();
30
31        System.out.println("-----minivan-----");
32
33        System.out.print("Enter passengers: ");
34        int p = input.nextInt();
35
36        System.out.print("Enter fuelcapacity: ");
37        int f = input.nextInt();
38
39        System.out.print("Enter mpg: ");
40        int m = input.nextInt();
41
42        minivan.passengers = p;
43        minivan.fuelcapacity = f;
44        minivan.mpg = m;
45    }
```

CIS 36A – 12th In Class / Take Home Assignment – 10 Points

Student Name	KaChi Lau	Student ID	10819338	Point Total
--------------	-----------	------------	----------	-------------

```

46 System.out.println("The minivan carries= " + minivan.passangers);
47 System.out.println("The minivan has a fuel capacity of= " + minivan.fuelcapacity);
48 System.out.println("The minivan mpg= " + minivan.mpg);
49 System.out.println("The minivan has a range of: " + minivan.range(minivan.fuelcapacity, minivan.mpg) + " miles");
50
51 //truck
52 Vehicle truck = new Vehicle();
53
54 System.out.println("-----truck-----");
55
56 System.out.print("Enter passangers: ");
57 p = input.nextInt();
58
59 System.out.print("Enter fuelcapacity: ");
60 f = input.nextInt();
61
62 System.out.print("Enter mpg: ");
63 m = input.nextInt();
64
65 truck.passangers = p;
66 truck.fuelcapacity = f;
67 truck.mpg = m;
68 System.out.println("The truck carries= " + truck.passangers);
69 System.out.println("The truck has a fuel capacity of= " + truck.fuelcapacity);
70 System.out.println("The truck mpg= " + truck.mpg);
71 System.out.println("The truck has a range of: " + truck.range(truck.fuelcapacity, truck.mpg) + " miles");
72
73 //motorcycle
74 Vehicle motorcycle = new Vehicle();
75
76 System.out.println("-----motorcycle-----");
77
78 System.out.print("Enter passangers: ");
79 p = input.nextInt();
80
81 System.out.print("Enter fuelcapacity: ");
82 f = input.nextInt();
83
84 System.out.print("Enter mpg: ");
85 m = input.nextInt();
86
87 motorcycle.passangers = p;
88 motorcycle.fuelcapacity = f;
89 motorcycle.mpg = m;
90 System.out.println("The motorcycle carries= " + motorcycle.passangers);
91 System.out.println("The motorcycle has a fuel capacity of= " + motorcycle.fuelcapacity);
92 System.out.println("The motorcycle mpg= " + motorcycle.mpg);
93 System.out.println("The motorcycle has a range of: " + motorcycle.range(motorcycle.fuelcapacity, motorcycle.mpg) + " miles");
94
95 }
96 }
97

```

```

run:
-----minivan-----
Enter passengers: 7
Enter fuelcapacity: 8
Enter mpg: 9
The minivan carries= 7
The minivan has a fuel capacity of= 8
The minivan mpg= 9
The minivan has a range of: 72 miles
-----truck-----
Enter passengers: 10
Enter fuelcapacity: 11
Enter mpg: 12
The truck carries= 10
The truck has a fuel capacity of= 11
The truck mpg= 12
The truck has a range of: 132 miles
-----motorcycle-----
Enter passengers: 13
Enter fuelcapacity: 14
Enter mpg: 15
The motorcycle carries= 13
The motorcycle has a fuel capacity of= 14
The motorcycle mpg= 15
The motorcycle has a range of: 210 miles
BUILD SUCCESSFUL (total time: 9 seconds)

```

Task 2 - Write a program that computes a single filer's income tax burden.

TAX RATE	Single Filers Income
10%	Up to \$6000
15%	\$6,001 - \$27,950
27%	\$27,951 - \$67,700
30%	\$67,701 - \$141,250
35%	\$141,251 - \$307,050
38.6%	\$307,051 or more

The user should be able input her income using **new** Scanner (System.in), input method and then be returned the amount of tax owed.

All source code for solving the problem and handling user input should be created in a “programmer created class.”

Use **return** for retrieving all values from calculations or **if/else** statements etc.

main will be used to operate the program.

Output should have proper formatting for dollars, 2 decimal places.

Sample Output- //Output should have proper formatting for dollars, 2 decimal places

Income tax for a single person making \$85000.00 is \$25500.00

Income tax for a single person making \$9800.00 is \$1470.00

```
1 package javaApplication2;
2
3
4 import java.util.Scanner;
5
6 class TaxRate {
7
8     void print(){
9
10         System.out.print("Please Enter your TaxRate: ");
11
12         Scanner input = new Scanner(System.in);
13         Double tax = input.nextDouble();
14
15         System.out.println("Income tax for a single person making $" + String.format("%.2f", tax) +
16             " is " + String.format("%.2f", yourtax(tax)));
17     }
18
19     public double yourtax(double newValue){
20         double tax = newValue;
21
22         if(tax < 0) {
23             System.out.println(" You entered a negative value!");
24         }
25
26         if(tax >= 0) {
27             if(tax < 6001) {
28                 tax = tax * 0.10;
29             }
30             if((tax >= 6001) && (tax <= 27950)) {
31                 tax = tax * 0.15;
32             }
33             if((tax >= 27951) && (tax <= 67700)) {
34                 tax = tax * 0.27;
35             }
36             if((tax >= 67701) && (tax <= 141250)) {
37                 tax = tax * 0.30;
38             }
39         }
40     }
41 }
```

CIS 36A – 12th In Class/ Take Home Assignment – 10 Points

Student Name **KaChi Lau** Student ID **10819338** Point Total

```

39         if((tax >= 141251) && (tax <= 307050)) {
40             tax = tax * 0.35;
41         }
42         if(tax > 307051) {
43             tax = tax * 0.386;
44         }
45     }
46     return tax;
47 }
48 }
49
50 public class JavaApplication1 {
51
52     public static void main(String[] args) {
53
54         TaxRate t = new TaxRate();
55
56         t.print();
57     }
58 }
59
60

```

run:

Please Enter your TaxRate: 9800

Income tax for a single person making \$9800.00 is 1470.00

BUILD SUCCESSFUL (total time: 3 seconds)

run:

Please Enter your TaxRate: 85000

Income tax for a single person making \$85000.00 is 25500.00

BUILD SUCCESSFUL (total time: 16 seconds)