

<http://www.javatpoint.com/this-keyword>

Define-

Encapsulation- Encapsulation is the packing of data and functions into a single component. The features of encapsulation are supported using classes in most object-oriented programming languages, although other alternatives also exist.

this.keyword- this is a reference to the current object — the object whose method or constructor is being called. You can refer to any member of the current object from within an instance method or a constructor by using this.

constructor chaining-

Calling another constructor in the same class from another constructor is called constructor chaining. By using this() we can call another constructor in the same class. Incase we want to call another constructor, this() should be the first line in the constructor. Below example shows code for constructor chaining. - See more at:

http://www.java2novice.com/java_constructor_examples/constructor_chaining/#sthash.sd4G8fGN.dpuf

Programming Assignments:

USE OBJECT ORIENTATED PROGRAM DESIGN TO SOLVE PROBLEM

Task 1- Create a class BMI. (Body Mass Index)

$BMI = 703 * weight / height * height$ [[height squared]]

The BMI class should also return the person's name using the program.

Create a method for calculating BMI and round BMI to 1 decimal place

Create a method for returning the BMI possibilities

Severe Thinness < 16

Moderate Thinness 16.1 – 17

Mild Thinness 17.1 – 18.5

Normal 18.6 – 26

Overweight 26.1 – 30

Obese Class I 30.1 – 35

Obese Class II 35.1 – 40

Obese Class III > 40.1

Output:

The BMI for John is 26.5 which is the Overweight Category.

or

The BMI for Susan is 17 which is in the Moderate Thinness Category

```
1
2 package javaapplication1;
3
4 import java.util.Scanner;
5
6 class BMI{
7     double weight, height, result;
8     String name;
9
10    public BMI(double h, double w, String n) {
11        height = h;
12        weight = w;
13        name = n;
14    }
15
16    public void calculate() {
17        result = ((weight * 703) / (height * height));
18    }
19
20    public void print() {
21
22        String p = "";
23        if (result < 16) {
24            p = "Severe Thinness";
25        } else if ((result >= 16.1) && (result <= 17)) {
26            p = "Moderate Thinness";
27        } else if ((result >= 17.1) && (result <= 18.5)) {
28            p = "Mild Thinness";
29        } else if ((result >= 18.6) && (result <= 26)) {
30            p = "Normal";
31        } else if ((result >= 26.1) && (result <= 30)) {
32            p = "Overweight";
33        } else if ((result >= 30.1) && (result <= 35)) {
34            p = "Obese Class I";
35        } else if ((result >= 35.1) && (result <= 40)) {
36            p = "Obese Class II";
37        } else if (result > 40.1) {
38            p = "Obese Class III";
```

```

39     }
40     System.out.println("The BMI for " + name + " is " +
41         (String.format("%.1f%n", result)) +
42         "which is the " + p + ".");
43 }
44
45 }
46
47 public class JavaApplication1 {
48
49     public static void main(String[] args) {
50
51         Scanner input = new Scanner(System.in);
52
53         System.out.print("Enter name: ");
54         String n = input.nextLine();
55         System.out.print("Enter weight: ");
56         double w = input.nextDouble();
57         System.out.print("Enter height: ");
58         double h = input.nextDouble();
59
60         BMI a = new BMI(h, w, n);
61
62         a.calculate();
63         a.print();
64
65     }
66 }
67
68 }
69

```

```

run:
Enter name: John
Enter weight: 15
Enter height: 16
The BMI for John is 41.2
which is the Obese Class III.
BUILD SUCCESSFUL (total time: 20 seconds)

```

USE OBJECT ORIENTATED PROGRAM DESIGN TO SOLVE PROBLEM

Task 2- Create a class Course.

Create 1 course. Fill the course with female and male students.

Output the course name and the total number of students enrolled, number of women enrolled and the names of students enrolled in the course.

Output:

CIS36A contains 10 students, 6 Female Students, and 4 males students.

Mary Smith, Lisa Brown, Jill Scott, Bobby Jones, etc.

```
1 package javaapplication2;
2
3
4 import java.util.Scanner;
5
6 class Course{
7     String [] total;
8     String [] female;
9     String [] male;
10    String n;
11
12    public Course(String[] m, String[] f, String courseName) {
13        male = m;
14        female = f;
15        total = new String[male.length + female.length];
16        n = courseName;
17    }
18
19    public void totalStudent() {
20
21        int i = 0;
22        while(i < female.length) {
23            total[i] = female[i]; //insert female first
24            i++;
25        }
26
27        i = female.length;
28        int j = 0;
29        while(i < total.length) {
30            total[i] = male[j]; //insert male
31            j++; i++;
32        }
33    }
34
35    public void print() {
36
37        System.out.println(n + " contains " + total.length + " students," +
38            female.length + " Female Students, and " + male.length +
39            " males students.");
40        for(int i = 0; i < 4; i++) {
41            System.out.print(total[i] + ", ");
42        }
43
44        System.out.println("etc.");
45    }
46
47 }
48
49 public class NewClass {
50
```

```
51 public static void main(String[] args) {  
52  
53     Scanner input = new Scanner(System.in);  
54  
55     String [] male = {"Ka Chi Lau", "King Jackson", "Bush Lee", "Jacky Chan"};  
56     String [] female = {"Mary Smith", "Lisa Brown", "Jill Scott",  
57         "Bobby Jones", "Kitty Li", "Micheal Johnson"};  
58  
59     System.out.print("Enter Course Name: ");  
60     String coursename = input.nextLine();  
61  
62     Course a = new Course(male, female, coursename);  
63     a.totalStudent();  
64     a.print();  
65  
66 }  
67  
68 }  
69
```

NewClass > main >

Output

JavaApplication1 (run) JavaApplication1 (run) #2

```
run:  
Enter Course Name: CIS36A  
CIS36A contains 10 students,6 Female Students, and 4 males students.  
Mary Smith, Lisa Brown, Jill Scott, Bobby Jones, etc.  
BUILD SUCCESSFUL (total time: 3 seconds)
```


CIS 27– 20th In Class / Lab Assignment – **10 Points**

Student Name	Student ID	Point Total
---------------------	-------------------	--------------------