

Key point: You can use nested **if** statements to write a program for computing taxes.

Table 3.2 2009 U.S. Federal Personal Tax Rates

Marginal Tax Rate	Single	Married Filing Jointly or Qualifying Widow(er)	Married Filing Separately	Head of Household
	Status=0	Status=1	Status=2	Status=3
10%	\$0-\$8,350	\$0-\$16,700	\$0-\$8,350	\$0-\$11,950
15%	\$8,351-\$33,950	\$16,701-\$67,900	\$8,351-\$33,950	\$11,951-\$45,500
25%	\$33,951-\$82,250	\$67,901-\$137,050	\$33,951-\$68,525	\$45,501-\$117,450
28%	\$82,251-\$171,550	\$137,051-\$208,850	\$68,526-\$104,425	\$117,451-\$190,200
33%	\$171,551-\$372,950	\$208,851-\$372,950	\$104,426-\$186,475	\$190,201-\$372,950
35%	\$372,951+	\$372,951+	\$186,475+	\$372,951+

Listing 3.5 ComputeTax.java

```

1  import java.util.Scanner;
2
3  public class ComputeTax {
4      public static void main(String[] args) {
5          //Create a Scanner
6          Scanner input = new Scanner(System.in);
7
8          //Prompt the user to enter filing status
9          System.out.print("(0-single filer, 1-married jointly or " +
10             "qualifying widow(er), 2-married separately, 3-head of " +
11             "household) Enter the filing status: ");
12
13         int status = input.nextInt();
14
15         //Prompt the user to enter taxable income
16         System.out.print("Enter the taxable income: ");
17         double income = input.nextDouble();
18
19         //Compute tax
20         double tax = 0;
21
22         if (status == 0) { //Compute tax for single filers
23             if (income <= 8350)
24                 tax = income * 0.10;
25             else if (income <= 33950)
26                 tax = 8350 * 0.10 + (income - 8350) * 0.15;
27             else if (income <= 82250)
28                 tax = 8350 * 0.10 + (33950 - 8350) * 0.15 +
29                     (income - 33950) * 0.25;
30             else if (income <= 171550)
31                 tax = 8350 * 0.10 + (33950 - 8350) * 0.15 +
32                     (82250 - 33950) * 0.25 + (income - 82250) * 0.28;

```

```
33         else if (income <= 372950)
34             tax = 8350 * 0.10 + (33950 - 8350) * 0.15 +
35                 (82250 - 33950) * 0.25 + (171550 - 82250) * 0.28 +
36                 (income - 171550) * 0.33;
37         else
38             tax = 8350 * 0.10 + (33950 - 8350) * 0.15 +
39                 (82250 - 33950) * 0.25 + (171550 - 82250) * 0.28 +
40                 (372950 - 171550) * 0.33 + (income - 372950) * 0.35;
41     } else if (status == 1) {           //Compute tax for married file jointly
42         if (income <= 16700)           //or qualifying widow(er)
43             tax = income * 0.10;
44         else if (income <= 67900)
45             tax = 16700 * 0.10 + (income - 16700) * 0.15;
46         else if (income <= 137050)
47             tax = 16700 * 0.10 + (67900 - 16700) * 0.15 +
48                 (income - 67900) * 0.25;
49         else if (income <= 208850)
50             tax = 16700 * 0.10 + (67900 - 16700) * 0.15 +
51                 (137050 - 67900) * 0.25 + (income - 137050) * 0.28;
52         else if (income <= 372950)
53             tax = 16700 * 0.10 + (67900 - 16700) * 0.15 +
54                 (137050 - 67900) * 0.25 + (208850 - 137050) * 0.28 +
55                 (income - 208850) * 0.33;
56         else
57             tax = 16700 * 0.10 + (67900 - 16700) * 0.15 +
58                 (137050 - 67900) * 0.25 + (208850 - 137050) * 0.28 +
59                 (372950 - 208850) * 0.33 + (income - 372950) * 0.35;
60     } else if (status == 2) {           //Compute tax for married file separately
61         if (income <= 8350)
62             tax = income * 0.10;
63         else if (income <= 33950)
64             tax = 8350 * 0.10 + (income - 8350) * 0.15;
65         else if (income <= 68525)
66             tax = 8350 * 0.10 + (33950 - 16700) * 0.15 +
67                 (income - 33950) * 0.25;
68         else if (income <= 104425)
69             tax = 8350 * 0.10 + (33950 - 16700) * 0.15 +
70                 (68525 - 33950) * 0.25 + (income - 68525) * 0.28;
71         else if (income <= 186475)
72             tax = 8350 * 0.10 + (33950 - 16700) * 0.15 +
73                 (68525 - 33950) * 0.25 + (104425 - 68525) * 0.28 +
74                 (income - 104425) * 0.33;
75         else
76             tax = 8350 * 0.10 + (33950 - 16700) * 0.15 +
77                 (68525 - 33950) * 0.25 + (104425 - 68525) * 0.28 +
78                 (186475 - 104425) * 0.33 + (income - 186476) * 0.35;
79
80
```

```
81         } else if (status == 3) {           //Compute tax for Head of Household
82             if (income <= 11950)
83                 tax = income * 0.10;
84             else if (income <= 45500)
85                 tax = 11950 * 0.10 + (income - 11950) * 0.15;
86             else if (income <= 117450)
87                 tax = 11950 * 0.10 + (45500 - 11950) * 0.15 +
88                     (income - 45500) * 0.25;
89             else if (income <= 190200)
90                 tax = 11950 * 0.10 + (45500 - 11950) * 0.15 +
91                     (117451 - 45500) * 0.25 + (income - 117450) * 0.28;
92             else if (income <= 372950)
93                 tax = 11950 * 0.10 + (45500 - 11950) * 0.15 +
94                     (117451 - 45500) * 0.25 + (190200 - 117450) * 0.28 +
95                     (income - 190200) * 0.33;
96             else
97                 tax = 11950 * 0.10 + (45500 - 11950) * 0.15 +
98                     (117451 - 45500) * 0.25 + (190200 - 117450) * 0.28 +
99                     (372950 - 190200) * 0.33 + (income - 372950) * 0.35;
100         } else {                          //Error – invalid status
101             System.out.println("Error – invalid status");
102             System.exit(1); //Terminates the program abnormally, 0 for normal
103         }
104     }
105
106     //Display result
107     System.out.println("Tax is " + (int)(tax * 100) / 100.0);
108 }
109 }
```

run:
(0-single filer, 1-married jointly or qualifying widow(er), 2-married separately, 3-head of household) Enter the filing status: 0
Enter the taxable income: 400000
Tax is 117683.5
BUILD SUCCESSFUL (total time: 28 seconds)