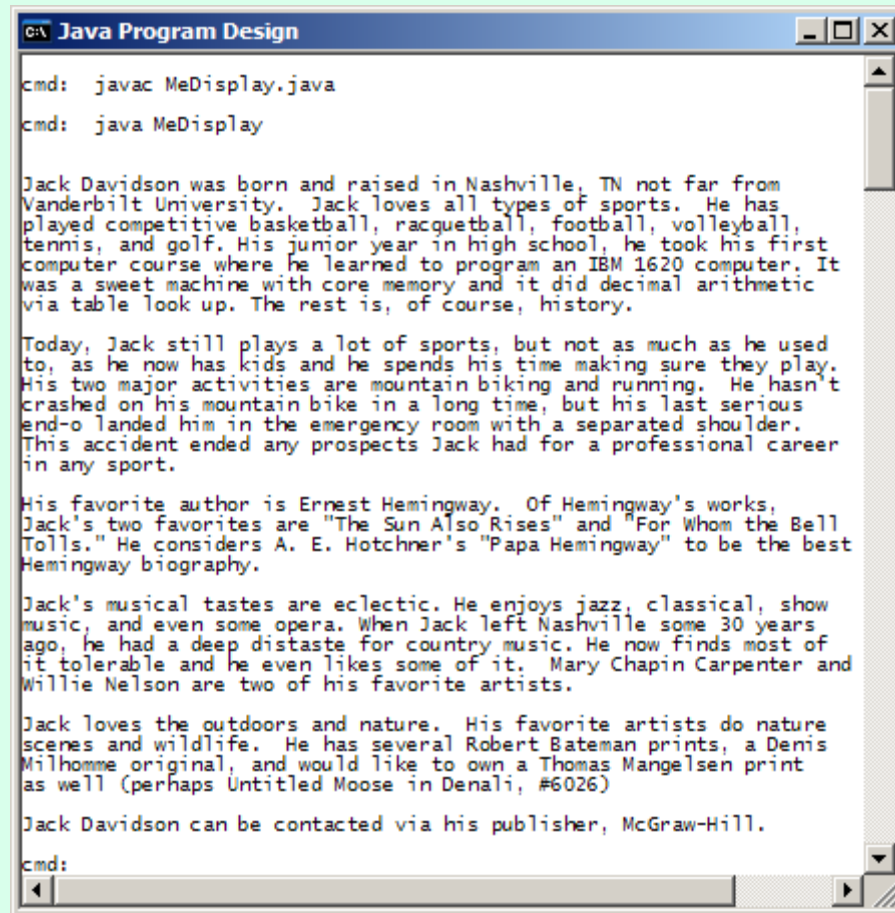


Figure 2.7 A MeDisplay compilation and run.

```
C:\ Java Program Design
cmd: javac MeDisplay.java
cmd: java MeDisplay

Jack Davidson was born and raised in Nashville, TN not far from
Vanderbilt University. Jack loves all types of sports. He has
played competitive basketball, racquetball, football, volleyball,
tennis, and golf. His junior year in high school, he took his first
computer course where he learned to program an IBM 1620 computer. It
was a sweet machine with core memory and it did decimal arithmetic
via table look up. The rest is, of course, history.

Today, Jack still plays a lot of sports, but not as much as he used
to, as he now has kids and he spends his time making sure they play.
His two major activities are mountain biking and running. He hasn't
crashed on his mountain bike in a long time, but his last serious
end-o landed him in the emergency room with a separated shoulder.
This accident ended any prospects Jack had for a professional career
in any sport.

His favorite author is Ernest Hemingway. Of Hemingway's works,
Jack's two favorites are "The Sun Also Rises" and "For Whom the Bell
Tolls." He considers A. E. Hotchner's "Papa Hemingway" to be the best
Hemingway biography.

Jack's musical tastes are eclectic. He enjoys jazz, classical, show
music, and even some opera. When Jack left Nashville some 30 years
ago, he had a deep distaste for country music. He now finds most of
it tolerable and he even likes some of it. Mary Chapin Carpenter and
Willie Nelson are two of his favorite artists.

Jack loves the outdoors and nature. His favorite artists do nature
scenes and wildlife. He has several Robert Bateman prints, a Denis
Milhomme original, and would like to own a Thomas Mangelsen print
as well (perhaps Untitled Moose in Denali, #6026)

Jack Davidson can be contacted via his publisher, McGraw-Hill.

cmd:
```

2.12**PROGRAMMING PROJECT — TRAINING ZONE**

There are two kinds of exercise—aerobic and anaerobic exercise. Normally, sustained activities such as bicycling, running or swimming are aerobic exercises. While activities requiring bursts of actions such as weight lifting and tennis are anaerobic exercises. Both types of exercise have their advantages. For example, anaerobic exercise can stimulate muscle growth and aerobic exercise can raise your metabolism by stimulating the production of fat-burning enzymes.

The objective of this case study is to practice interactive programming in the context of problem solving

For a sustained activity to be an aerobic exercise there needs to be an elevation in the heart rate. However, the elevation cannot be extreme. If the elevation is extreme, then muscles are oxygen deprived and they burn sugars rather than fat. Research indicates that there is a heart beat rate training zone that should be kept in order to get the maximum aerobic effect from an exercise. The actual training zone for a person is based on many factors, such as the individual's normal heart rate, fitness, health, weight, etc. However, there are formulas indicating reasonable training zones for the majority of fit people with normal heart rates. The formula used in this section comes from HealthCentral.com. The formula work as follows: subtract the age of interest from 220; 65% of that value is the low end of the training zone range and 80% of that value is the high end of the training zone range.

The goal of this case study is to develop a program that computes a training zone according to the following problem statement.

Provide an interactive training zone calculator that prompts a user for her or his age. The calculator program then computes and displays the training zone for that age. The low end of the training zone is 65% of the difference between 220 and the specified age. The high end of the training zone is 80% of the difference between 220 and the specified age.



ANALYSIS AND DESIGN

Before determining the necessary information structures and algorithm, you should depict a sample input/output behavior. A depiction should help clarify what information is being provided and what information is to be computed. Furthermore, if the software is being developed for a client, then it also aids you in verifying with the client that the proper problem is being solved. An example of a possible input/output behavior for the training zone calculator is given in Figure 2.8.

Figure 2.8 A possible training zone calculator console window display.

A screenshot of a Java Program Design console window. The window title is 'C:\ Java Program Design'. The text inside the window shows the following sequence of commands and output:

```
cmd: javac TrainingZone.java
cmd: java TrainingZone
A heart been training zone is calculated
for your age of interest.
Enter age (number): 20
Suggested heart beat training zone for
a fit person of 20 years with a normal
heart rate is 130 to 160 beats per minute.
cmd:
```

An analysis of the problem statement indicates one piece of information that must be gotten from the user—the age of interest. From that information and the training zone formula you derive the lower and upper heart beat rates for the training zone associated with that age of interest. An analysis of the problem also indicates that in developing an algorithm three integer variables are appropriate: *ageOfInterest*, *lowBeatRate*, and *highBeatRate*. In addition, three constants are appropriate: *BASE_RATE* with value 220, *LOW_ZONE_MULTIPLIER* with value 65%, and *HIGH_ZONE_MULTIPLIER* with value 80%.

From the preceding discussion, an algorithm for solving the problem is relatively easy.

- Step 1.* Provide a suitable legend for the calculator.
- Step 2.* Prompt the user for the age for which they desire a suggested training zone.
- Step 3.* Extract the user-specified age and assign it to variable *ageOfInterest*.
- Step 4.* Compute the low heart beat rate by assigning *lowBeatRate* the value of the expression $LOW_ZONE_MULTIPLIER \times (BASE_RATE - ageOfInterest)$.
- Step 5.* Compute the high heart beat rate by assigning *highBeatRate* the value of the expression $HIGH_ZONE_MULTIPLIER \times (BASE_RATE - ageOfInterest)$.
- Step 6.* Display the training zone information as defined by *ageOfInterest*, *lowBeatRate*, and *highBeatRate*.



Implementation

Producing a Java program `TrainingZone.java` from the suggested algorithm can be done nicely with a single class `TrainingZone`. The class should be documented and laid out appropriately. For example, the code should make use of program header and method comments, whitespace, and a consistent indentation scheme.



TESTING

One way to test a program informally is to do a hand-check—compute by hand what the output should be for several different ages. Then run the program using those same ages and compare the results. Determine for any discrepancies whether the program or hand calculation is incorrect. If the program is incorrect, review the formulas already given and your implementation of them. Also consider adding temporarily statements displaying intermediary results in an effort to determine where your program's correctness breaks down. (Such statements are called *debugging statements*.) Testing and debugging is examined in detail in Chapter 13.