



A short Java program is listed below. One block of the program is missing. Your challenge is to **match the candidate block of code** (on the left) **with the output** that you'd see if the block were inserted. Not all the lines of output will be used, and some of the lines of output might be used more than once. Draw lines connecting the candidate blocks of code with their matching command-line output.

```
class Test {
    public static void main(String [] args) {
        int x = 0;
        int y = 0;
        while (x < 5) {
            
            System.out.print(x + " " + y + " ");
            x = x + 1;
        }
    }
}
```

Candidate code goes here

Candidates:

`y = x - y;`

`y = y + x;`

```
y = y + 2;
if( y > 4 ) {
    y = y - 1;
}
```

```
x = x + 1;
y = y + x;
```

```
if ( y < 5 ) {
    x = x + 1;
    if ( y < 3 ) {
        x = x - 1;
    }
}
y = y + 2;
```

Possible output:

22 46

11 34 59

02 14 26 38

02 14 36 48

00 11 21 32 42

11 21 32 42 53

00 11 23 36 410

02 14 25 36 47

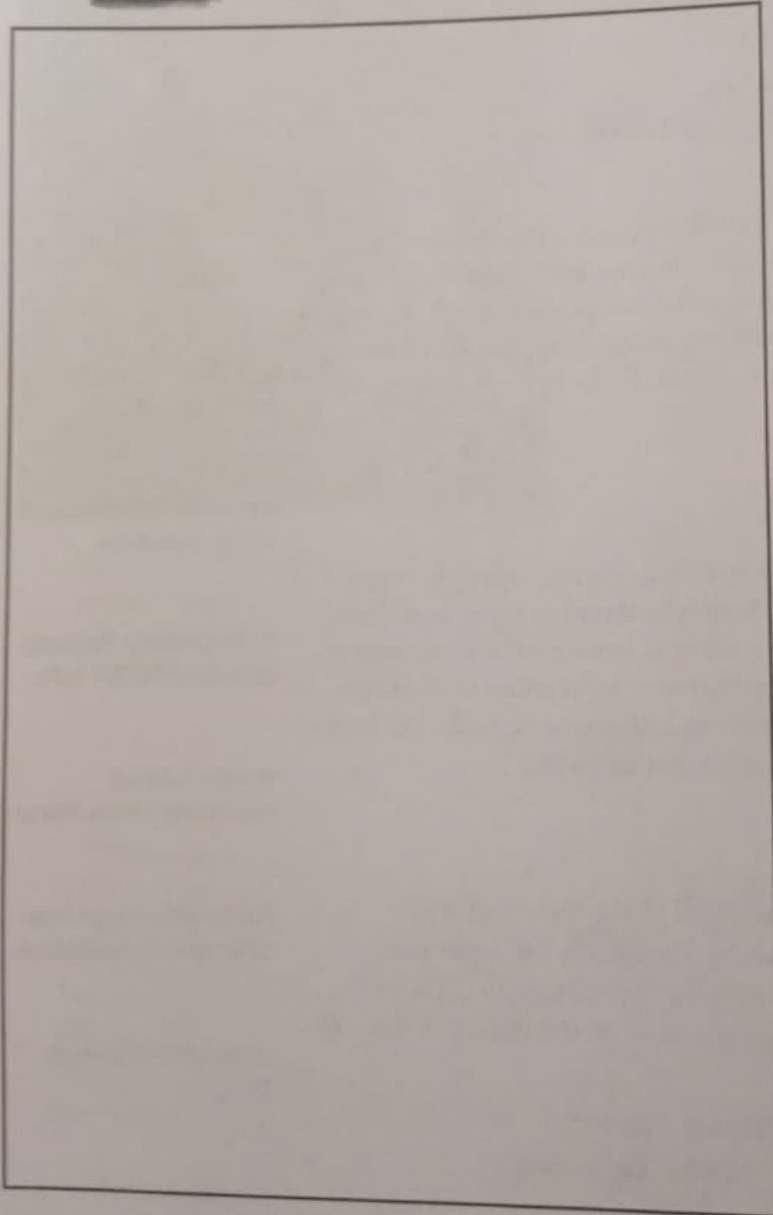
Match each candidate with one of the possible outputs

Answers on page 26.



Code Magnets

A working Java program is all scrambled up on the fridge. Can you rearrange the code snippets to make a working Java program that produces the output listed below? Some of the curly braces fell on the floor and they were too small to pick up, so feel free to add as many of those as you need!



Output:

```
File Edit Window Help Sleep
% java Shuffle1
a-b c-d
```

```
if (x == 1) {
    System.out.print("d");
    x = x - 1;
}
```

```
if (x == 2) {
    System.out.print("b c");
}
```

```
class Shuffle1 {
    public static void main(String [] args)
```

```
if (x > 2) {
    System.out.print("a");
}
```

```
int x = 3;
```

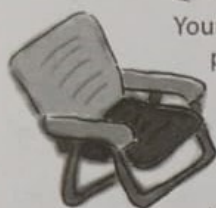
```
x = x - 1;
System.out.print("-");
```

```
while (x > 0) {
```

puzzle: Pool Puzzle



Pool Puzzle



Your **job** is to take code snippets from the pool and place them into the blank lines in the code. You may **not** use the same snippet more than once, and you won't need to use all the snippets. Your **goal** is to make a class that will compile and run and produce the output listed. Don't be fooled—this one's harder than it looks.

→ Answers on page 26.

Output

```
File Edit Window Help Cheat
%java PoolPuzzleOne
a noise
annoys
an oyster
```

Note: Each snippet from the pool can be used only once!

```
System.out.print(" ");
System.out.print("a");
System.out.print("\n");
System.out.print("an");
```

```
x > 0
x < 1
x > 1
x > 3
x < 4
```

```
x = x + 1;
x = x + 2;
x = x - 2;
x = x - 1;
```

```
System.out.print("noys ");
System.out.print("oise ");
System.out.print(" oyster ");
System.out.print("annoys");
System.out.print("noise");
```

```
class PoolPuzzleOne {
    public static void main(String [] args) {
        int x = 0;

        while ( _____ ) {

            _____
            if ( x < 1 ) {
                _____
            }

            if ( _____ ) {
                _____
            }

            _____
        }
        if ( x == 1 ) {
            _____
        }
        if ( _____ ) {
            _____
        }
        System.out.println();
    }
}
```



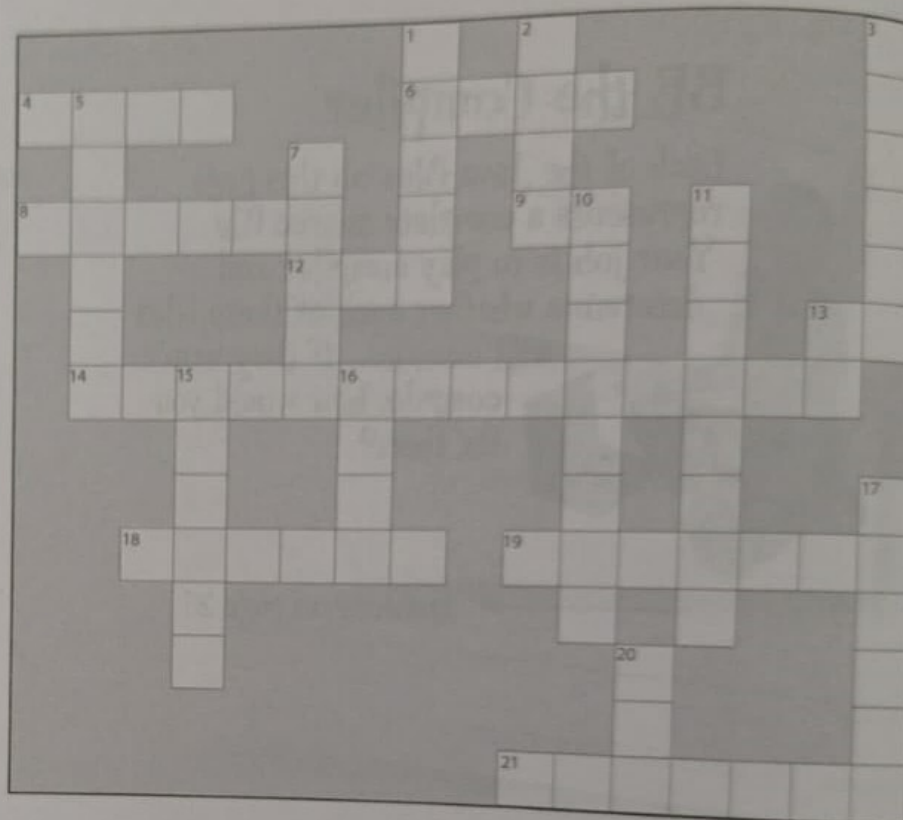
JavaCross

Let's give your right brain something to do.

It's your standard crossword, but almost all of the solution words are from Chapter 1. Just to keep you awake, we also threw in a few (non-Java) words from the high-tech world.

Across

4. Command line invoker
6. Back again?
8. Can't go both ways
9. Acronym for your laptop's power
12. Number variable type
13. Acronym for a chip
14. Say something
18. Quite a crew of characters
19. Announce a new class or method
21. What's a prompt good for?



Down

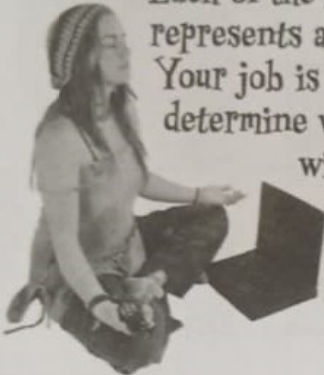
1. Not an integer (or ____ your boat)
2. Come back empty-handed
3. Open house
5. 'Things' holders
7. Until attitudes improve
10. Source code consumer
11. Can't pin it down
13. Department for programmers and operations
15. Shocking modifier
16. Just gotta have one
17. How to get things done
20. Bytecode consumer

→ Answers on page 26.



BE the Compiler

Each of the Java files on this page represents a complete source file. Your job is to play compiler and determine whether each of these files will compile. If they won't compile, how would you fix them?



—————> Answers on page 25.

B

```
public static void main(String [] args) {
    int x = 5;
    while ( x > 1 ) {
        x = x - 1;
        if ( x < 3 ) {
            System.out.println("small x");
        }
    }
}
```

A

```
class Exercisela {
    public static void main(String[] args) {
        int x = 1;
        while (x < 10) {
            if (x > 3) {
                System.out.println("big x");
            }
        }
    }
}
```

C

```
class Exerciselc {
    int x = 5;
    while (x > 1) {
        x = x - 1;
        if (x < 3) {
            System.out.println("small x");
        }
    }
}
```