1 **import** java.util.Random;

2 **import** java.util.concurrent.ArrayBlockingQueue;

3

4 **public** **class** ConsumerV3 **implements** Runnable

5 {

6 ArrayBlockingQueue <Integer> sharedData;

7 **int**[] timesConsumed = **new** **int**[10];

8 **boolean**[] consumedData = **new** boolean[10];

9

10 **public** ConsumerV3(ArrayBlockingQueue <Integer> sharedData)

11 {

12 **this**.sharedData = sharedData;

13 }

14

15 **public** **void** run()

16 {

17 Random delay = **new** Random();

18 Integer dataItem = 0;

19

20 **for**(**int** i = 1; i <= 10; i++)

21 {

22 **try**

23 {

24 Thread.sleep(delay.nextInt(10) + 1); **//simulate data fetch**

25 dataItem = (Integer) sharedData.take();

26 System.out.println("Consumed " + dataItem + " <---");

27 }

28 **catch**(InterruptedException e)

29 {

30 }

31

32 **//record consumed statistics**

33 consumedData[dataItem - 1] = true;

34 timesConsumed[dataItem - 1]++;

35 }

36 outputConsumedSummary();

37 }

38

39 **private** **void** outputConsumedSummary()

40 {

41 **try**

42 {

43 Thread.sleep(5000);

44 }

45 **catch**(InterruptedException e)

46 {

47 }

48 System.out.print("Consumed data: ");

49 **for**(int i = 1; i <= 10; i++)

50 {

51 **if**(consumedData[i-1] == **true**)

52 {

53 System.out.print(" " + i);

54 }

55 }

56 System.out.print("\nTimes consumed:");

57 **for**(**int** i = 1; i <= 10; i++)

58 {

59 **if**(consumedData[i-1] == true)

60 {

61 System.out.print(" " + timesConsumed[i-1]);

62 }

63 }

64 }

65 }

**Figure 14.27 The class ConsumerV3.**