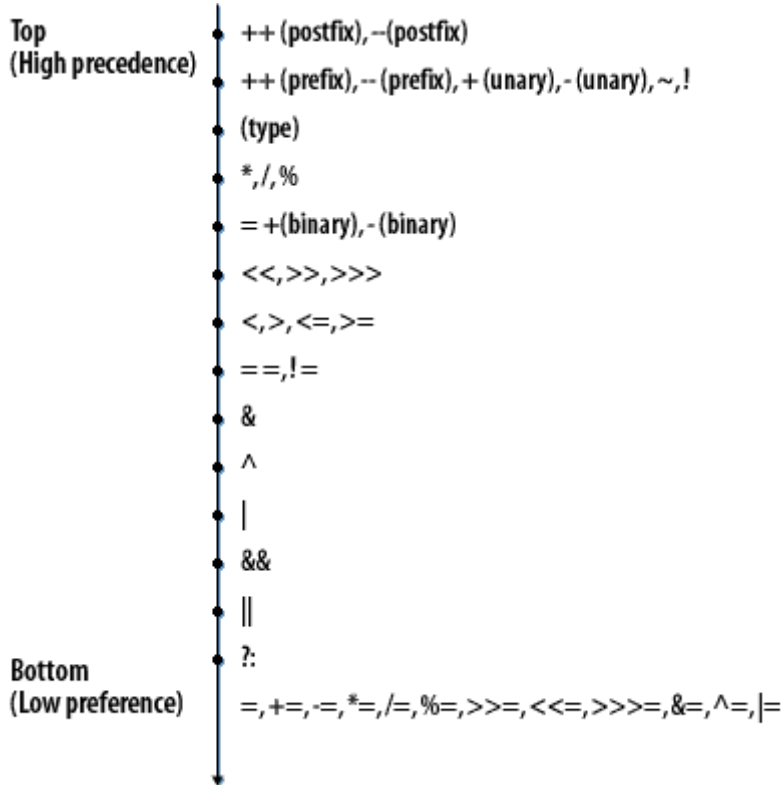


1.?

Terminalden yazılacak evde dene

2.C ve E

Javada operatör önceliği şekildeki gibidir. Şekil incelendiğinde C ve E seçenekleri doğrudur.



3.B, C, D, F

Belirli işlevleri yerine getirmek için kullanılan, tek başına çalışamayan ve içindeki private property leri public methodlarla get ve set eden java classlarına verilen isim.

Get metodunun değişken almaması gerektiği için A seçeneği yanlıştır. B ve C seçeneğindeki getter setter kullanımı doğrudur. “is” ile başlayan metot isimleri de boolean da kullanılması uygundur. Bu yüzden D seçeneği yanlışken F seçeneği doğrudur. gimmeEars() rastgele bir metot gibi duruyor.

4.A,D

```
public class Q4 {  
    public static void main(String[] args) {  
        int[] crossword[] = new int[10][20];  
        for (inti = 0; i < crossword.length; i++)  
            for (intj = 0; j < crossword.length; j++)  
                crossword[i][j] = 'x';  
        System.out.println(crossword.length);  
        //crossword.length seklindedüzeltilmesigerekliyor.  
    }  
}
```

5.B,D,

Uygulama error verdiğinde sonlanacağı için mümkün oldukça vermemesi üzerine yazılmalıdır. (B)

Try catch bloğu ile hata yakalanabilir.(D)

6.A,C,D

Java.lang.object bir şey yapmadan da erişilebilir.(D)

Savana.sand e erişilebilir ama sand'ın altsınıflarına erişilemez.(B)

7.C

Mutable yeni nesne yaratmadanda değerleri değiştirilebilenlerdir, immutable de tam tersi. Array ve StringBuilder mutabileyken diğerleri değildir.

8.C

```
public class Q8 {
    public static void main(String[] args) {
        StringBuilder builder = new StringBuilder("Leaves growing");
        do {
            builder.delete(0, 5);
        } while (builder.length() > 5);
        System.out.println(builder);
    }
}
```

9.C

```
package reality;
public class Equivalency {
    public static void main(String[] edges) {
        final String ceiling = "up";
        String floor = new String("up");
        final String wall = new String(floor);
        System.out.print((ceiling == wall) + " " + (floor == wall) + " " +
            ceiling.equals(wall));
    }
}
```

10.C

```
public class Giggles {
    public static void main(String[] args) {
        String lol = "lol";
        System.out.println(lol.toUpperCase() == lol);
        System.out.println(lol.toUpperCase() == lol.toUpperCase());
        System.out.println(lol.toUpperCase().equals(lol));
        System.out.println(lol.toUpperCase().equals(lol.toUpperCase()));
        System.out.println(lol.toUpperCase().equalsIgnoreCase(lol));
        System.out.println(lol.toUpperCase().equalsIgnoreCase(lol.toUpperCase()));
    }
}
```

11.A,B,C

İnner ve outer olmadan da aynı sonucu vermektedir(A)

12.B,C

Long değer ondalık ifade kabul etmiyor.

13.A

```
import java.time.*;
public class OnePlusOne {
    public static void main(String... nums) {
        LocalDateTime time = LocalDateTime.of(1, 11);
        while (time.getHour() < 1) {
            time.plusHours(1);
            System.out.println("in loop");
        }
    }
}
```

14.D

```
package fun;

public class Sudoku {
    static int[][] game;

    public static void main(String args[]) {
        game[3][3] = 6;
        //üsteki satırı yüzünden NullPointerException
        Object[] obj = game;
        obj[3] = 'X';
        System.out.println(game[3][3]);
    }
}
```

15.C,E ama A da hata vermiyor sor??

16.?

17.C

f.getFish() metodundan dolayı derlenmeyecektir.

18.A

kod çıktısı -5 tir. Yani bir satırlık çıktı verir.

19.F

Try olmadan tek başına kullanılamayacak catch ve finally ifadeleri var ama bunların ikisinin de olduğu şık bulunmadığı için none of above oluyor.

20.A

```
public class Q20 {
    public static void main(String[] args) {
        int result = 8;
        loop: while (result > 7) {
            result++;
            do {
                result--;
            } while (result > 5);
            break loop;
        }
        System.out.println(result);
    }
}
```

21.C

Kod çalıştırıldığında 1 2 çıktısını vermektedir.

22.A

```
public class Costume {  
    public static void main(String[] args) {  
        String witch = "p";  
        String tail = "lack";  
        witch.concat(tail);  
        System.out.println(witch);  
    }  
}
```

23.A,C,F

<https://docs.oracle.com/javase/tutorial/java/>

The interface body can contain **abstract methods**, **default methods**, and **static methods**. An abstract method within an interface is followed by a semicolon, but no braces (an abstract method does not contain an implementation). Default methods are defined with the default modifier, and static methods with the static keyword. **All abstract, default, and static methods in an interface are implicitly public, so you can omit the public modifier.**

In addition, an interface can contain constant declarations. All constant values defined in an interface are implicitly public, static, and final. Once again, you can omit these modifiers.

24.D

`while (tie = null)` ifadesinde “=” ifadesi tek kullanıldığı için kod derlenmeyecektir.

25.B,F

26.C

m1 satırından dolayı kod derlenememektedir.

27.D

`profit$$$` ifadesi initialize edilmediği için hata vermektedir.

28.A

29.B,C,E

Trouble(), Trouble(int deep) ve Trouble(Long deep) constructors larında compile hatası bulunmaktadır.

30.E,F

Statik olmayan değişkenlerde “Cannot make a static reference to the non-static field max” hatası alınır.

31.??

32.C

```

public class Legos {
    public static void main(String[] args) {
        StringBuilder sb = new StringBuilder();
        sb.append("red");
        sb.deleteCharAt(0);
        sb.delete(1, 1);
        System.out.println(sb);
    }
}

```

33.A,D

Javada metod isimleri küçükharf, "\$" ve "_", ile başlayabilirler. Sadece bu kurala uyan A ve D seçeneğidir.

34.??

35.F

Java ile ilgili verilen 2,3,4 seçenekleri doğrudur ama hepsini kapsayan bir şık yoktur.

36.C

```

public class Q36 {
    public static void main(String[] args) {
        String[][] listing = new String[][] { { "Book", "34.99" }, { "Game", "29.99" },
        { "Pen", ".99" } };
        System.out.println(listing.length + " " + listing[0].length);
    }
}

```

37.A, B, E

<https://docs.oracle.com/javase/tutorial/java/nutsandbolts/switch.html>

The switch Statement

Unlike if-then and if-then-else statements, the switch statement can have a number of possible execution paths. A switch works with the byte, short, char, and int primitive data types. It also works with enumerated types (discussed in Enum Types), the String class, and a few special classes that wrap certain primitive types: Character, Byte, Short, and Integer (discussed in Numbers and Strings).

38.D

"d" değişkeni tanımlanmadığı için kod derlenmeyecektir.

39.A,B,E

40.A,F

41.D

```

import java.util.*;

public class Museums {
    public static void main(String[] args) {
        String[] array = { "Natural History", "Science", "Art" };
        List<String> museums = Arrays.asList(array);
        museums.remove(2);
    }
}

```

```

//calıstırıldığında yukarıdaki satırdan dolayı hata aklinıyor. kaynak bulamamasorunu
System.out.println(museums);
}
}

```

42.A,D

```

public class Underscores {
    public String _name = "Sherrin";

    public void massage() {
        int zip = 10_0_17;
    }
}

```

43.B

```

package counting;
import java.util.*;
public class Binary {
    public static void main(String[] args) {
        args = new String[] { "0", "1", "01", "10" };
        Arrays.sort(args);
        System.out.println(Arrays.toString(args));
    }
}

```

44.??

45.A

Kod çıktısı OCA OCAOCA OCPOCP OCAOCP OCP sekinde tek satırdırç.

46.C,D

Javac komutu ile .java dosyasını derleriz ve .class uzantılı dosya elde ederiz. Java komutu ile de programı çalıştırırız.

47.C

```

public class Q47 {
    public static void main(String[] args) {
        char one = Integer.parseInt("1");
        Character two = Integer.parseInt("2");
        int three = Integer.parseInt("3");
        Integer four = Integer.parseInt("4");
        short five = Integer.parseInt("5");
        Short six = Integer.parseInt("6");
    }
}

```

Sadece 3 ve 4. Satırlar derlenebilir.

48.B,D,F

Eğer int olursa sonuç 5 dönmektedir.

Eğer long olursa sonuç 2 dönmektedir. boolean olursa derlenememektedir.

49.A

```

import java.time.*;

```

```

public class Equality {
    public void main(String[] args) {
        System.out.println(new StringBuilder("zelda") == new StringBuilder("zelda"));
        System.out.println(3 == 3);
        System.out.println("bart" == "bart");
        System.out.println(newint[0] == newint[0]);
        System.out.println(LocalTime.now() == LocalTime.now());
    }
}

```

Uygulama çıktısı “5” tir yani hiç “true” yazdırmaz.

50.D

Kod çıktısı”245”tir.

51.E

Kod derleniyor anca çalıştırınca exception veriyor.

52.??

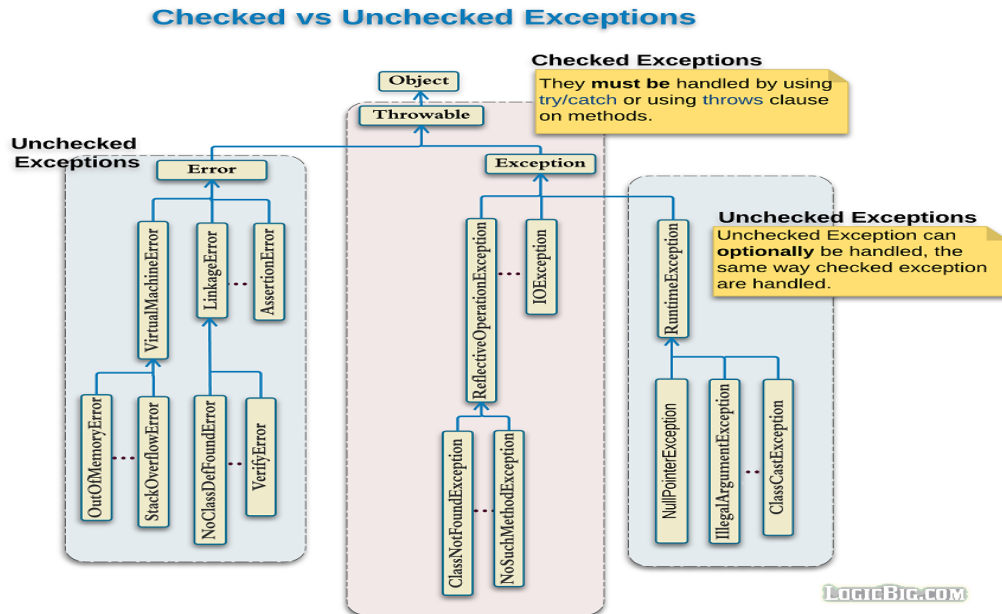
53.C,D,E

10 ve 12.satırlar silinmeli ve default olan 11 veya 14. Satırlardan birisi silinmelidir.

54.A,B,C

h1,h2,h3 satırlarından dolayı kod derlenmemektedir.

55. E,F,B



56.F

Kod çıktısı

“Goodbye
See you again” şeklindedir.

57.??

58.??

59.C

6. ve 8. Satırlarda compile error vardır.

60. A,E

61.C

```
public class Q61 {  
    public static void main(String[] args) {  
        int fish = 1 + 2 * 5 >= 2 ? 4 : 2;  
        int mammals = 3 < 3 ? 1 : 5 >= 5 ? 9 : 7;  
        System.out.print(fish + mammals + "");  
    }  
}
```

62.A ..??

63. terminalden çalıştır. Kodunu yazdım

64.??

65.E

Fill in the blanks: The **throws** keyword is used in method declarations, the **finally** keyword is used to guarantee a statement will execute even if an exception is thrown, and the **throw** keyword is used to throw an exception to the surrounding process.

66.B,E

Kod çıktısı aşağıdaki gibidir. B ve E seçeneklerini doğru yaomaktadır.

```
Downtown Day  
Uptown Night  
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: 2  
    at nyc.TouristBus.main(TouristBus.java:9)
```

67.E

Because of **virtual methods** , it is possible to **override** a method, which allows Java to support **polymorphism**.

68.E

```
package calendar;  
  
public class Seasons {  
    public static void seasons(String... names) {  
        int l = names[1].length(); // s1  
        System.out.println(names[1]); // s2  
    }  
}
```



```

public static void main(String[] args) {
    seasons("Summer", "Fall", "Winter", "Spring");
}
}

```

69.D

```

package percussion;
interface MakesNoise {
}

abstract class Instrument implements MakesNoise {
    public Instrument(int beats) {
    }

    public void play() {
    }
}

public class Drum extends Instrument {
    public void play(int count) {
    }

    public void concert() {
        super.play(5);
    }

    public static void main(String[] beats) {
        MakesNoise mn = new Drum();
        mn.concert();
    }
}

```

70.B

71.C

Üç yerde compile error vermektedir.

72.D ??

73.D

String ifadesi küçük harfle yazıldığı için hata verecektir.

74.terminalden çalıştırma evde dene kodu yazdım

75.??

76.E

77.??

78.F

Compile errorlar bulunmaktadır.

79.D

10,11,12. Satırlarda compile error

80.??

Default ifadesini araştır