***Ejercicio 1:***

1. Es válido.
2. No es válido: comienza con número.
3. Es válido.
4. No es válido; palabra reservada.
5. Es válido.
6. Es válido.
7. No es válido: contiene espacio vacío.
8. Es válido.
9. No es válido: operador “- “.
10. Es válido.
11. No es válido: contiene “ ’ ”.
12. Es válido.
13. No es válido: comienza con un número, contiene “#”.
14. Es válido.
15. No es válido: contiene operador “%”.
16. Es válido.
17. Es válido.
18. Es válido.
19. Es válido.
20. No es válido: símbolo “.”.
21. Es válido.
22. No es válido: símbolo “?”.
23. Es válido.
24. No es válido: palabra reservada.
25. Es válido.
26. Es válido.
27. Es válido.
28. Es válido.
29. No es válido: comienza con número.
30. No es válido: contienen símbolo “&”.
31. No es válido: comienza con número.
32. No es válido: palabra reservada.

***Ejercicio 2:***

1. Es válido. Es tipo double escrito en decimal.
2. Es válido. Es tipo double escrito en decimal.
3. Es válido. Es tipo double escrito en decimal.
4. Es válido. Es tipo double escrito en decimal.
5. Es válido. Es tipo int escrito en decimal.
6. No es válido. Posición inválida del guión bajo.
7. No es válido. No puede haber decimales en el exponente.
8. No es válido. No puede haber espacios en blanco.
9. Es válido. Es tipo int escrito en octal.
10. No es válido. Comienza con cero, pero luego se expresa en hexadecimal.
11. Es válido. Es tipo int escrito en hexadecimal.
12. Es válido. Es tipo int en hexadecimal.
13. Es válido. Es tipo long en decimal.
14. No es válido. Posición inválida del guión bajo.
15. Es válido. Es tipo int en octal.
16. Es válido. Es tipo int en octal.
17. No es válido. Más de un punto decimal.
18. Es válido. Es tipo float en decimal.
19. Es válido. Es tipo int en binario.
20. Es válido. Es tipo long en decimal.
21. Es válido. Es tipo int en hexadecimal.
22. Es válido. Es tipo long en hexadecimal.
23. No es válido. Posición incorrecta del guión bajo.
24. No es válido. Posición incorrecta del prefijo 0B.
25. Es válido. Es tipo int en hexadecimal.
26. Es válido. Es tipo double en decimal.
27. No es válido. Falta el exponente.
28. Es válido. Es tipo float en decimal.
29. No es válido. Posición incorrecta del guión bajo.
30. No es válido. Comienza como binario, pero luego se expresa en hexadecimal.
31. Es válido. Es tipo int en hexadecimal.
32. Es válido. Es tipo long en hexadecimal.
33. No es válido. Falta 0X.
34. Es válido. Es tipo int en octal.
35. Es válido. Es tipo double en decimal.
36. Es válido. Es tipo double en decimal.
37. No es válido. Posición incorrecta de guión bajo.
38. Es válido. Es tipo double en decimal.
39. No es válido. Falta el exponente.
40. No es válido. Un binario sólo admite ceros y unos.
41. Es válido. Es tipo int en decimal.
42. Es válido. Es tipo int en hexadecimal.
43. Es válido. Es tipo int en hexadecimal.
44. Es válido. Es tipo double en decimal.
45. No es válido. Debería comenzar con 0B.
46. No es válido. Más de un punto decimal.
47. Es válido. Es tipo float en decimal.
48. No es válido. La letra “G” no pertenece al sistema hexadecimal.

***Ejercicio 3:***

1. Es válido.
2. Es válido.
3. Es válido.
4. No es válido. Más de un carácter no de escape.
5. Es válido.
6. No es válido. Más de un carácter.
7. No es válido. Deben ser comillas simples.
8. Es válido.
9. No es válido. Más de un carácter.
10. Es válido.
11. No es válido. Más de un carácter, comillas dobles.
12. Es válido.
13. Es válido.
14. Es válido.
15. No es válido. Fuera de comillas simples.
16. No es válido. Duera de comillas simples.
17. Es válido.
18. Es válido.
19. Es válido.
20. Es válido.

***Ejercicio 4:***

1. No es válido. Sin comillas dobles.
2. Es válido.
3. No es válido. Comillas dobles sin cerrar.
4. Es válido.
5. Es válido.
6. Es váldio
7. Es válido.
8. Es válido.
9. No es válido. Comillas dobles sin cerrar.
10. Es válido.
11. No es válido. Comillas dobles sin cerrar y sin secuencia de escape.
12. No es válido. Sin comillas dobles.
13. No es válido. Comillas simples.
14. Es váldio.
15. Es válido.

***Ejercicio 5:***

*Aclaración: por el editor de texto, el principio de cada declaración aparece en mayúscula: corresponde minúscula.*

Parte 1:

1. Int p = 22, q=-23;
2. Float x=0.33f, y=23.556f, z=3435.09f;
3. Char a=’m’, b=’z’, c=’s’;
4. Double raiz1=1.121, raiz2=78.23;
5. Long contador=0;
6. Short indicador=2;
7. Int índice=21;
8. Double precio=23.44, precioFinal=44.77;
9. Char car1=’g’, car2=’p’;
10. Byte valor=6;
11. Boolean primero=true, ultimo=false;
12. String nombre=”pedro”;

Parte 2:

1. int edad=28;
2. int postalCode=5513;
3. double altura=1.79;
4. char genero=’H’;
5. string nombre=”Maxi”;
6. int hijos=3;
7. float iva=20.5f;
8. int tallaCamisa=44;
9. double peso=88.0;
10. float precio=23.66f;
11. string alumnoRepetidor=”NO”;
12. string mensaje=”hola”;
13. char letra=’g’;
14. int mayorEdad=70;
15. int minutos=23;
16. int dias=55;
17. string matriculaAuto=”AF GT 5678”;
18. int contador=0;
19. boolean mayorDeEdad=true;
20. string tallaCamiseta=”XL”;

***Ejercicio 6:***

1. 3/2+4/3
2. 1/(x-5)-3\*x\*y/4
3. 1/2+7
4. 7+1/2
5. a\*a/(b-c)+(d-e)/(f-g\*h/j)
6. m/n+p
7. m+n/(p-q)
8. a\*a/(b\*b)+c\*c/(d\*d)
9. (m+n/p)/(q-r/s)
10. (3\*a+b)/(c-(d+5\*e)/(f+g/(2\*h))
11. (a\*a+2\*a\*b+b\*b)/((1/(x\*x))+2)

***Ejercicio 7:***

Parte 1:

1. -> 161.5
2. -> No están definidas en este ejercicio.
3. -> “d” no está definida.
4. -> No definidos.
5. -> No definidos
6. -> No definidos
7. -> No definidos

Parte 2:

x= 88, y = 3.5, z = -5.2

1. 86.3
2. 286.6
3. 25.1429
4. 0.5
5. -51.7647
6. 19.9429
7. 205.3333
8. 16.7619
9. -4.0
10. -149.6
11. 93.2
12. 0.2
13. 91.2
14. 94.9

Parte 3:

1. 70
2. 69-53+63=79
3. 51
4. 3
5. 98
6. 6
7. 100
8. 82.0189
9. 159
10. 2703

***Ejercicio 8:***

a=8, b=3, c=-5

1. 6
2. 45
3. 2.6667
4. 2
5. -1.6
6. -2
7. -4.8
8. -4.8
9. 2
10. 1
11. 18
12. 0.2
13. 0.6922

***Ejercicio 9:***

1. Int
2. Float
3. Double
4. Double
5. Float
6. Int
7. Double
8. Int
9. Double
10. Int
11. Int
12. Int
13. Double
14. Long
15. Int
16. Double
17. Double
18. Float
19. Long
20. Float
21. Double
22. Int
23. Int
24. Int
25. int
26. Double

***Ejercicio 10:***

Parte 1:

1. True
2. True
3. True
4. False
5. True
6. True

Parte 2:

1. False
2. True
3. True
4. True
5. True
6. False

***Ejercicio 11:***

1. False
2. False
3. True
4. False
5. True
6. True
7. True
8. False
9. False
10. True
11. True
12. True
13. True
14. False
15. True
16. True
17. False
18. True
19. True
20. False
21. False
22. True

***Ejercicio 12:***

Parte 1:

Lo que se imprime en consola es:

10

11

12

12

13

13

15

16

19

19

21

Parte 2:

Lo que se imprime en consola es:

c

d

f

f

d

f

h

f

h

e

Parte 3:

1. i=2
2. i=1
3. i=3
4. i=2
5. i=3
6. i=3
7. j=0
8. i=4
9. i=3
10. k=3

***Ejercicio 13:***

Básico inicial 1:

El programa creado fue:

public class Main {

  public static void main(String[] args) {

    int N=4;

    double A=23.76;

    char C='y';

    System.out.println(N);

    System.out.println(A);

    System.out.println(C);

    System.out.println(N+A);

    System.out.println(A-N);

    System.out.println(0+C);

  }

}

El output en consola fue:

4

23.76

y

27.76

19.76

121

Básico inicial 2:

El programa creado fue:

public class Main {

  public static void main(String[] args) {

    int X=9, Y=6;

    double N=1.34, M=-34.7;

    System.out.println(X+" "+Y+" "+N+" "+M);

    System.out.println(X+Y);

    System.out.println(X-Y);

    System.out.println(X\*Y);

    System.out.println(X/Y);

    System.out.println(X%Y);

    System.out.println(N+M);

    System.out.println(N-M);

    System.out.println(N\*M);

    System.out.println(N/M);

    System.out.println(N%M);

    System.out.println(X+N);

    System.out.println(Y/M);

    System.out.println(Y%M);

    System.out.println(X\*2);

    System.out.println(Y\*2);

    System.out.println(N\*2);

    System.out.println(M\*2);

    System.out.println(X+Y+N+M);

    System.out.println(X\*Y\*N\*M);

  }

}

El output fue de:

9 6 1.34 1.34 -34.7

15

3

54

1

3

-33.36

36.040000000000006

-46.498000000000005

-0.03861671469740634

1.34

10.34

-0.17291066282420747

6.0

18

12

2.68

-69.4

-18.360000000000003

-2510.8920000000003

Básico inicial 3:

El programa creado fue:

public class Main {

  public static void main(String[] args) {

    int N=22;

    N=N+77;

    System.out.println(N);

    N=N-3;

    System.out.println(N);

    N=N\*2;

    System.out.println(N);

  }

}

El output fue:

99

96

192

Básico inicial 4:

El programa creado fue:

public class Main {

  public static void main(String[] args) {

    int A=4, B=56, C=-123, D=-4;

    System.out.println(A+" "+B+" "+C+" "+D);

    B=C;

    C=A;

    A=D;

    D=B;

    System.out.println(A+" "+B+" "+C+" "+D);

  }

}

El output fue:

4 56 -123 -4

-4 -123 4 -123

Básico inicial 5:

El código planteado fue:

public class Main {

  public static void main(String[] args) {

    int A=-12;

    System.out.println(A + (A%2==0 ? " es par " : " es impar "));

  }

}

El output:

-12 es par

Básico inicial 6:

El programa es:

public class Main {

  public static void main(String[] args) {

    int A=-990;

    System.out.println(A + (A<0 ? " es negativo " : " es positivo o nulo "));

  }

}

El output:

-990 es negativo

Básico inicial 7:

El programa:

public class Main {

  public static void main(String[] args) {

    int C=-123;

    System.out.println(C + (C<0 ? " es negativo " : " es positivo "));

    System.out.println(C + (C%2==0 ? " es par " : " es impar "));

    System.out.println(C + (C%5==0 ? " es múltiplo de 5 " : " no es múltiplo de 5 "));

    System.out.println(C + (C%10==0 ? " es múltiplo de 10 " : " no es múltiplo 10 "));

    System.out.println(C + (C<100 ? " es menor que 100 " : " no es menor que 100 "));

  }

}

El output:

-123 es negativo

-123 es impar

-123 no es m�ltiplo de 5

-123 no es m�ltiplo 10

-123 es menor que 100

***Ejercicio 16:***

Parte 1:

1. k=i=8
2. k=j=5
3. z=x=0.005F
4. z=0
5. a=b=’p’
6. k=j=5
7. k=0
8. k=9
9. a=c=’q’
10. a=’r’

Parte 2:

1. x=2
2. x=4
3. x=10
4. x=4
5. x=6
6. x=2
7. x=-1
8. x=0
9. x=5
10. x=0

***Ejercicio 17:***

1. M=N\*2;
2. M=N\*3;
3. P=6\*(A-B);
4. A\*B-(C+D);
5. M=(A%10)/2;
6. D\_1=(N%10);

N=N/10;

D\_2=(N%10);

N=N/10;

D\_3=N%10;

S=D\_1+D\_2+D\_3;

1. D\_1=N%10;

D\_2=M%10;

S=D\_1+D\_2;

1. B=(N%2==0) && (N%3==0);

System.out.println( N + (B ? “ Es múltiplo de 2 y de 3 “ : “ no es múltiplo de 2 ni de 3 ”));

1. D\_1=N%10;

System.out.println( N + (D\_1%2==0 ? “ Es par“ : “ no es impar ”));

1. D\_1=N/100;

System.out.println( D\_1 + (D\_1%2!=0 ? “ Es impar “ : “ no es impar ”));

1. D\_1=N/1000;

System.out.println( N + (D\_1%2==0 ? “ Es par “ : “ no es par ”));

1. (A >= ’A’ && A <= ’Z’)
2. (A >= ‘A’ && A <= ‘Z’ || A>=’a’ && A <= ‘z’ )
3. (A < ‘A’ || A > ‘Z’)
4. (A < ’A’ || A > ’Z’) && (A < ’a’ || A > ’z’)
5. (N %10 == 0 || N%7 == 0)
6. (precio >= 10 && precio < 50)
7. N+=77
8. M-=3
9. x\*=2
10. N=N+N%10 (sólo si N es de dos dígitos)
11. N/1000 == N%10 && N/100%10 == N/10%10
12. M == 4 || M == 6 || M == 9 || M == 11
13. N = N/10
14. N = N % 10000
15. C >= ‘0’ && C <= ‘9’
16. X = N/1000 + N/100%10 + N/10%10 + N%10
17. Y = M/1000 + M/100%10 + M/10%10 + M%10
18. Z = X + Y
19. AA – AN > 18 || AA – AN == 18 && MA > MN || AA – AN == 18 && MA == MN && DA >= DN

***Ejercicio 17:***

1. N / 1000 % 2 == 0
2. N / 10000 == N / 1000 % 10
3. N / 10000 == N % 10 && N / 1000 % 10 == N / 10 % 10
4. C == ‘A’ || C == ‘E’ || C == ‘I’ || C == ‘O’ || C == ‘U’)
5. !(C == ‘A’ || C == ‘E’ || C == ‘I’ || C == ‘O’ || C == ‘U’)
6. (A == ‘a’ || A == ‘e’ || A == ‘i’ || A == ‘o’ || A == ‘u’) &&

B == ‘a’ || B == ‘e’ || B == ‘i’ || B == ‘o’ || B == ‘u’)

1. A % 4 == 0 && A % 100 != 0 || A % 400 == 0
2. N / 10 == N % 10
3. (N / 10 + N % 10) % 2 == 0
4. N / 10 % 10 > N / 100 &&  N / 10 % 10 > N % 10
5. N / 1000 == 4 || N / 100 % 10 == 4 || N / 10 % 10 == 4 || N % 10 == 4
6. X = N / 100
7. N = N – N % 10 o también N -= N % 10
8. M = M - N % 10 o también M -= N % 10
9. N = N + (N > 5 ? 5 : 100)
10. N = N < 0 ? 5 : 100