Level Up OOP
Week 3: Momentum Builds
C# Vocabulary and Concepts
Ch. 4 - 6

[T] 1. Array (p.27)
[P] 2. dot operator (p.28)
[OO] 3. Methods (p.28)
[Z] 4. Properties (p.28)
[V] 5. Arguments (p.29)
[K] 6. Copy() (p.29)
[LL] 7. Length (p.29)
[H] 8. IndexOf() (p.30)
[M] 9. Sort() (p.30)
[O] 10. Concatenate (p.31)
[BB] 11. Substring() (p.32)
[U] 12. Equals() (p.33)
[SS] 13. Split() (p.33)
[KK] 14. Add() (p.35)
[J] 15. Count (p.36)
[E] 16. Insert() (p.36)
[II] 17. Remove() (p.36)
[S] 18. RemoveAt() (p.36)
[R] 19. Clear() (p.37)
[CC] 20. Contains() (p.37)

[Q] 21. reference type (p.37)
[PP] 22. value type (p.37)
[RR] 23. Write() (p.39)
[GG] 24. Console (p.40)
[C] 25. Placeholders (p.42)
[EE] 26. format specifier (p.43)
[MM] 27. Backslash (p.45)
[AA] 28. Newline (p.46)
[JJ] 29. Tab (p.45)
[N] 30. ReadLine() (p.46)
[W] 31. Convert (p.47)
[X] 32. ToInt32() (p.47)
[A] 33. ToDecimal() (p.48)
[UU] 34. ToDouble() (p.48)
[B] 35. ToSingle() (p.48)
[I] 36. Comparison (p.51)
[Y] 37. condition statement (p.51)
[TT] 38. control flow statement (p.51)
[FF] 39. logical operators (p.52)
[QQ] 40. Break (p.57)

[G] 41. Default (p.57)
[NN] 42. jump statement (p.57)
[HH] 43. Continue (p.65)
[DD] 44. Try-catch-finally (p.66)
[L] 45. Exception (p.68)
[D] 46. FormatException (p.69)
[F] 47. IndexOutOfRangeException (p.69)

Answer Bank

A. Used to convert to a decimal
B. Converts the value of the specified Decimal to the equivalent single-precision floating-point number
C. Curly braces that indicate where values are to be passed in to and concatenate with strings
D. A class that handles the specific error that occurs when the format of an argument is invalid
E. Method to add members at a specific position of a list
F. A class that handles the specific error that occurs when you try to access an element of an array with an index that is outside its bounds
G. An optional statement that is executed if no other case in a switch statement applies
H. Method to determine if a certain value exists in an array
I. ==; to compare whether two variables are the same
J. Property to find out the number of elements in a list
K. Method that allows you to copy the contents of one array into another array, starting from the first element
L. A class that handles general errors
M. Method used to sort array
N. Used to accept input from users; reads a line of characters
O. To join two strings
P. An operator used when we want to access a property or method of a class
Q. A data type that stores a reference to the data
R. Method to remove all items in a list
S. Method to remove a member at a specific location
T. A collection of data that are normally related to each other
U. Method to compare if two strings are identical
V. Data passed into a method
W. Used to convert data from one data type to another
X. Used to a 32-bit signed integer
Y. A statement that evaluates to true or false
Z. Represents a class's data
AA. \n; Used to add a newline to text in code
BB. Method used to extract a string from a longer string
CC. Method to check if a list contains a certain member
DD. Controls how the program proceeds when an error occurs
EE. Used to specify the format of numeric data in placeholders
FF. Useful to combine multiple condition statements

GG. Class that represents the standard input, output, and error streams for console applications
HH. A jump statement that instructs the compiler to exit the current block
II. Method to remove members from a list
JJ. \t; Used to add a tab to text in code
KK. Method to add members to a list
LL. Property of an array; tells the number of items the array has
MM. \; Used to escape characters that otherwise have a different meaning
NN. A statement that instructs the compiler to jump to another line in the program
OO. Represent a class's behavior
PP. A data type that stores its own data
QQ. A jump statement that instructs the compiler to skip the rest of the loop thereafter
RR. Method to display messages to users
SS. Method to split a string into substrings based on an array of user-defined separators
TT. A statement that determines whether or not other statements will be evaluated
UU. Converts the value of the specified Decimal to the equivalent double-precision floating-point number