For each of the prompts, write the answer to the prompt in Python on the left side, write the answer to the prompt in Javascript on the right side. The best way to do this is to write a small program in the appropriate IDE, then copy/paste the *tested and working* code into the appropriate box.

1. Declare a Variable with a value. Reassign a variable’s value

|  |  |
| --- | --- |
| x = 12  x = "My New Value" | let x = 12  x = "My New Value" |

1. Assign the result of an expression to a variable.

|  |  |
| --- | --- |
| x = 12  y = 36  z = x \* y | let x = 12  let y = 36  let z = x \* y |

1. Change the data type of a variable’s value and assign it to the original value
   1. Convert to string
   2. Convert to number (integer or float)

|  |  |
| --- | --- |
| x = 12  x = str(x) | let x = 12  x = x.toString() |
| x = "My New Value"  x = len(x) | let x = "My New Value"  x = x.length |

1. Create a conditional statement that branches depending on if a variable’s value is greater than or equal to 7, greater than 4, or neither

|  |  |
| --- | --- |
| x = 12  if x==7:  print('this number')  elif x>4:  print('that number')  else:  print('no number') | let x = 12  if (x==7){      console.log('this number');  } else if (x>4) {      console.log('that number');  } else {      console.log('no number');  } |

1. Create a conditional statement that only branches if the variable’s value is ‘blue’ OR ‘green’

|  |  |
| --- | --- |
| x = 'green'  if x=='blue' or x=='green':  print('awesome') | let x = 'green'  if (x=='blue' || x=='green') {      console.log('awesome');  } |

1. Create a conditional statement that only branches if the variable’s value is ‘black’ AND ‘yellow’

|  |  |
| --- | --- |
| x = 'black'  if x=='black' and x=='yellow':  print('awesome') | let x = 'black'  if (x=='black' && x=='yellow') {      console.log('awesome');  } |

1. Create a function that outputs “Hello” to the console

|  |  |
| --- | --- |
| def statement():  print('Hello')  statement() | function statement(){      console.log('hello')  };  statement() |

1. Create a function that returns “Hello”

|  |  |
| --- | --- |
| def statement():  x='Hello'  return x | function statement(){      let x='hello'      return x  }; |

1. Create a function that takes two arguments, adds their values together, assigns the new value to a local variable, and returns that variable

|  |  |
| --- | --- |
| def statement(x,y):  sentence = x + y  return sentence | function statement(x,y){      let sentence = x + y      return sentence  }; |

1. Create a loop that repeats while a variable’s value is true

|  |  |
| --- | --- |
| x=True  while x == True:  print("this will never end") | let x = true  while (x == true) {      console.log('this will never end')  } |

1. Create a loop that repeats while a variable’s value is true, but will break mid-loop if another variable equals false

|  |  |
| --- | --- |
| x=12  y=True  z=0  while x == 12 and y==True:  print("this will never end")  z += 1  if z>x:  y = False | let x = 12  let y = true  let z = 0  while (x == 12 && y==true) {      console.log('this will never end')      z+=1      if (z>x) {          y = false      }  } |

1. Create an array/list that contains three strings

|  |  |
| --- | --- |
| x=['one','two','three'] | let x = ['one','two','three'] |

1. Using the array from #12, remove the last element in the array

|  |  |
| --- | --- |
| x=['one','two','three']  x.remove('three') | let x = ['one','two','three']  x.splice(2,1) |

1. Using the array from #12, remove the first element in the array

|  |  |
| --- | --- |
| x=['one','two','three']  x.remove('one') | let x = ['one','two','three']  x.splice(0,1) |

1. Using the array from #12, remove the element in the middle (index 1)

|  |  |
| --- | --- |
| x=['one','two','three']  x.remove('two') | let x = ['one','two','three']  x.splice(1,1) |

1. Using the array from #12, output the first element

|  |  |
| --- | --- |
| x=['one','two','three']  print(x[0]) | let x = ['one','two','three']  console.log(x[0]) |

1. Create an empty array. Then add a string element to it

|  |  |
| --- | --- |
| x=[]  x.append('word') | let x = []  x.push('word'); |

1. Using the array from #12, use a loop to output each element

|  |  |
| --- | --- |
| x=['one','two','three']  for i in x:  print(i) | let x = ['one','two','three']  for (let i = 0; i < x.length; i++) {      console.log(x[i])  }; |

1. Using the array from #12, use a loop to output the index of each element

|  |  |
| --- | --- |
| x=['one','two','three']  for i in x:  print(x.index(i)) | let x = ['one','two','three']  for (let i = 0; i < x.length; i++) {      let y = x[i]      console.log(x.indexOf(y))  }; |

1. Create an object with one property and one method. The property value will be a string, and the method will output the value of this object’s property.

|  |  |
| --- | --- |
| class newClass:  x = "This value"  def classval():  print(x)    y = newClass() | const newClass = {      x : "This value",      classval : function(){          console.log(x)      }  } |

1. Using the object in #20, print the name of each property/method key in the object

|  |  |
| --- | --- |
| class newClass:  x = "This value"  def classval():  print(x)    y = newClass()  attributes = dir(y)  for attribute in attributes:  print(attribute) | const newClass = {      x : "This value",      classval : function(){          console.log(x)      }  }  console.log(Object.keys(newClass)) |

1. Create a variable whose value is a string equal to the name (key) of the property of the object in #20. Output the property value of the object by using the variable (not the property name)

|  |  |
| --- | --- |
| class newClass:  x = "This value"  def classval():  print(x)  y = newClass()  g="x"  print(getattr(y, g)) | const newClass = {      x : "This value",      classval : function(){          console.log(x)      }  }  let g="x"  console.log(newClass[g]) |

1. Create a two-dimensional array
   1. Create an array/list with three elements
   2. Each element is another array/list with three elements whose values are strings

|  |  |
| --- | --- |
| x = [['one','two','three'],['four','five','six'],['seven','eight','nine']] | let x = [['one','two','three'],['four','five','six'],['seven','eight','nine']] |

1. Using the array in #23, use two loops (one inside the other) to output all 9 elements in the multi-dimensional array.

|  |  |
| --- | --- |
| x = [['one','two','three'],['four','five','six'],['seven','eight','nine']]  for a in x:  for i in a:  print(i) | let x = [['one','two','three'],['four','five','six'],['seven','eight','nine']]  for (let number of x){      for (let integer of number){          console.log(integer)      };  }; |

1. Add a new property to the object used in #20 with a number value

|  |  |
| --- | --- |
| class newClass:  x = "This value"  y = 286  def classval():  print(x)    z = newClass() | const newClass = {      x : "This value",      y : 286,      classval : function(){          console.log(x)      }  } |