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

|  |  |
| --- | --- |
| number = 5  number = 10 | var number = 5;  var number = 10; |

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

|  |  |
| --- | --- |
| number = number + 5 | var number = number + 5; |

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)

|  |  |
| --- | --- |
| number = str(number)  number = int(number) | var number = String(number);  var number = Number(number); |

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

|  |  |
| --- | --- |
| if number >= 7:      print("number is greater than or equal to 7.")  elif number > 4:      print("number is greater than 4.")  else:      print("you failed.") | if (number >= 7){      console.log("number is greater than or equal to 7.")  }  else if(number > 4){      console.log("number is greater than 4.")  }  else{      console.log("you failed.")  }; |

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

|  |  |
| --- | --- |
| if color == "blue" or color == "green":      print("color is either blue or green.") | if (color == "blue" || color == "green"){      console.log("color is either blue or green.")  } |

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

|  |  |
| --- | --- |
| if color == "black" and color2 == "yellow":      print("colors are black and yellow") | if (color == "black" && color2 == "yellow"){      console.log("colors are black and yellow.")  }; |

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

|  |  |
| --- | --- |
| def speech():      print("Hello") | function speech(){      console.log("Hello")  }; |

1. Create a function that returns “Hello”

|  |  |
| --- | --- |
| def returnSpeech():      return "Hello" | function returnSpeech(){      return "Hello";  }; |

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 addition(x, y):      addedNumber = x + y      return addedNumber | function addition(x, y){      var addedNumber = x + y;      return addedNumber;  }; |

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

|  |  |
| --- | --- |
| while number:      print("number is True.") | while (number){      console.log("number is True.");  }; |

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

|  |  |
| --- | --- |
| while number:      print("number is True.")      if color == False:          break | while (number){      console.log("number is True.");      if (color == false){          break;      };  }; |

1. Create an array/list that contains three strings

|  |  |
| --- | --- |
| items = ["monitor", "mouse", "keyboard"] | var items = ["monitor", "mouse", "keyboard"]; |

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

|  |  |
| --- | --- |
| items.pop() | items.pop(); |

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

|  |  |
| --- | --- |
| items.pop(0) | items.shift(); |

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

|  |  |
| --- | --- |
| items.pop(1) | items.splice(1, 1); |

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

|  |  |
| --- | --- |
| print(items[0]) | console.log(items[0]); |

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

|  |  |
| --- | --- |
| people = []  people.append("alex") | people = [];  people.push("alex"); |

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

|  |  |
| --- | --- |
| for equipment in items:      print(equipment) | for (equipment in items){      console.log(items[equipment]);  }; |

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

|  |  |
| --- | --- |
| for equipment in items:      print(items.index(equipment)) | for (equipment in items){      console.log(equipment);  }; |

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 Actions:      def \_\_init\_\_(self):          self.talk = "Hello there."      def speak(self):          print(self.talk)  person = Actions() | const person = {    talk : "Hello there.",    speak: function() {      console.log(this.talk)    },  }; |

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

|  |  |
| --- | --- |
| print(dir(person)) | console.log(person); |

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)

|  |  |
| --- | --- |
| guy = person.talk  print(guy) | guy = person.talk;  console.log(guy); |

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

|  |  |
| --- | --- |
| recipes = [["eggs", "milk", "flour"], ["pumpkin", "sugar", "apple"], ["beef", "salt", "pepper"]] | recipes = [["eggs", "milk", "flour"], ["pumpkin", "sugar", "apple"], ["beef", "salt", "pepper"]]; |

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

|  |  |
| --- | --- |
| for needs in recipes:      for food in needs:          print(food) | for (x = 0; x <= 2; x++){      for (y = 0; y <= 2; y++){          console.log(recipes[x][y])      };  }; |

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

|  |  |
| --- | --- |
| class Actions:      def \_\_init\_\_(self):          self.talk = "Hello there."          self.bonesBroken = 2      def speak(self):          print(self.talk) | const person = {    talk : "Hello there.",    bonesBroken : 2,    speak: function() {      console.log(this.talk)    },  }; |