1. For each statement, indicate whether it is true or false in the blank provided:

1.1 Value types in TypeScript include number, array, and boolean.

1.2 Reference types hold a copy of the value they are assigned.

1.3 Class definitions hold all of the properties that an object has.

1.4 Properties can be declared and initialized within a class definition.

1.5 If there are three if then statements that are **NOT** nested, then only one boolean test expression will be evaluated.
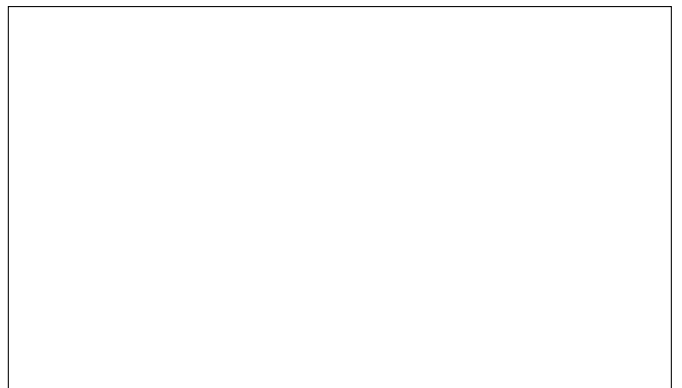
1.6 Objects of the same class don't have to have all of the properties defined in that class.

1.7 The new keyword is used to create a new object

1.8 A number array called arrNums is declared in main and then passed into a void function that multiplies all its elements by two. Back in main after the function is called, the elements in arrNums will be doubled even though the function was void and did not return the array.

2. Create a class with the following characteristics:

- Name: OfficeEmployee
- Properties:
    - name (type: `string`, initial value: "Jim")
    - position (type: `string`, initial value: "Salesman")
    - salary (type: `number`, initial value: 56000)
    - fullTime (type: `boolean`, initial value: true)
    - likeability (type: `number`, intial value: 10)
    - qualified (type: `boolean`, initial value: true)

3. Write a function called `sumEven` to add together all of the even values in a number array. The number array is passed in as a parameter named `arr`. Return the sum. Ensure that your code is not modifying the array that is being passed to it.

4. Call the functions below so that the value returned is the value asked for.

```
let insert = (str: string, inserted: string, index: number): string => {
    let out = "";
    for (let i = 0; i < index; i++) {
        out += str[i];
    }
    out += inserted;
    for (let i = index; i < str.length; i++) {
        out += str[i];
    }
    return out;
};

let reverse = (str: string): string => {
    let out = "";
    for (let i = str.length - 1; i >= 0; i--) {
        out += str[i];
    }
    return out;
};
```

4.1 Make one or more function calls (all on the same line) that will return "dlrowolleH"

4.2 Using your answer from the previous question, make one or more function calls (all on the same line)that will return "!dlrowolleH"

4.3 Using your answer from the previous question, make one or more function calls (all on the same line) that will return "Helloworld!!"

5. Use the following class to answer the questions.

```
1  class Burger {
2      meat: string = "hamburger";
3      bread: string = "white";
4      cheese: boolean = true;
5      pickles: number = 1;
6  }
```

5.1 Declare and initialize a new Burger object named myBurger.

5.2 Change the pickles property and the cheese property to be different from the default values.

6. Given the Animal objects to the left, answer the questions that follow.

```
1   let dog: Animal = new Animal();
2   dog.bark = true;
3   dog.hair = true;
4   dog.hairColor = "white";
5   dog.legs = 4;
6   dog.wings = false;
7   let dragon: Animal = new Animal();
8   dragon.bark = false;
9   dragon.hair = false;
10  dragon.legs = 4;
11  dragon.wings = true;
12  let person: Animal = new Animal();
13  person.bark = false;
14  person.hair = true;
15  person.hairColor = "blonde";
16  person.legs = 2;
17  person.wings = false;
```

6.1 Based on the given properties, define the `Animal` class

6.2 Create an `Animal` named `horse` that has the properties of a horse with brown hair.

7. The following function has been written with poorly named variables, which makes it harder to read and potentially harder to debug. After you have decided what this function accomplishes, rename the function as well as all of the variables to make it easier to read. Note: There isn't just one correct name for each variable.

```
1   let foo = (arr: number[]): number => {
2       let w = 0;
3       let x = 0;
4       for (let i = 0; i < arr.length; i++) {
5           let y = arr[i] < 0;
6           if (y) {
7               w += arr[i];
8               x++;
9           }
10      }
11      let z = w / x;
12      return z;
13  };
```

7.1 What is a better name for `foo`?

7.2 What is a better name for `w`?

7.3 What is a better name for `x`?

7.4 What is a better name for `y`?

7.5 What is a better name for `z`?