Name _____

Q.1 Write the outputs for the following code-blocks (executed independently)
Also mention output / return values for typeof statements.

```var a = [1,2,3];```
```a.shift();```
```console.log(a);```	

```function f1(n){```
 	
    ```function f2(N) {console.log(n/N)); }```
    ```return f2;```
 	
```}```
 	
```var N = 4;```
```var n = 2;```
```var x = f1(N);```
```console.log (x(n));```	

```var x = 2;```
```x=5;```
```function f(){ "use strict"; console.log(this.x);}```
```f();```	

```var a = 5;```
```var c = "5";```
```var b = new String(a);```
```var d = new String(c);```
 	
```typeof a;```
```typeof b;```
```typeof c;```
```typeof d;```	

```console.log(window.document == document);```	
```var d = new Date(0);``` ```console.log(d.getFullYear());```	
```var a = "nagarro";``` ```console.log(a.toUpperCase());```	

```var abs = Math.abs;```
```var b = 4;```
```console.log(b.abs());```	

```console.log(Math.SQRT2);```	

``
var b = 4;
if (true) {
var c = 5;
let d = 6;
const e = 7;
 	
}
 	
console.log(b);
console.log(c);
console.log(d);
console.log(e);
``	

`( function (n) { console.log("Coding" + n); } )(2);`	

``
let a = {name : "A"};
a.name = "B";
console.log(a.name);
const b = { name : "C"};
b.name = "D";
console.log(b.name);
``	

``
var b = "Workshop JavaScript";
console.log(b.trim());
``	

``
var a = "String";
var b = new String("string");
typeof a;
typeof b;
console.log(a==b);
console.log(a ===b);
``	

Q.2 To manually coerce a String object to a string primitive type; what method to use?

Q.3 What is wrapping (in context of string and String) ? What are the benefits of wrapping?

Q.4 Will the following code raise any errors? Why/Why not?

```
var a = "nagarro";
console.log(a.toUpperCase());
```

Q.5 Write code for a function factory makeAdder that returns an AdderN function given an integer N as a parameter The function AdderN (given N is an integer) takes an integer input p and behaves as follows:

AdderN(p) returns N+p as the output
Eg.

Adder5(4) returns 9
Adder5(11) returns 16
Adder10(4) returns 14

So write code for makeAdder that returns Adder5, Adder10 … AdderN given N as input.

Q.6 Differentiate amongst let, const and var in the context of ES6