EXAMPLE 29: STRINGS AND NUMBERS -BUT NOT OBJECTS - ARE 'PASSED-BY-VALUE' TO FUNCTIONS

EXAMPLE 29: STRINGS AND NUMBERS -BUT NOT OBJECTS - ARE 'PASSED-BY-VALUE' TO FUNCTIONS

IN OTHER WORDS, A COPY IS MADE OF STRING OR NUMBER VARIABLES

AND THE FUNCTION WORKS WITH THE COPY - NOT THE ORIGINAL.

EXAMPLE 29: STRINGS AND NUMBERS -BUT NOT OBJECTS - ARE 'PASSED-BY-VALUE' TO FUNCTIONS

IN OTHER WORDS, A COPY IS MADE OF STRING OR NUMBER

AND THE FUNCTION WORKS WITH THE COPY - NOT THE ORIGINAL.

EVEN IF THE FUNCTION MODIFIES THE VARIABLE, THE ORIGINAL IS UNCHANGED.

EVEN IF THE FUNCTION MODIFIES THE functival such that the function of the functival such that the function is the function of the func

```
var x = 10;
 var y = "Vitthal";
 console.log("Initial values (in calling function) = " + x + " " + y);
 modifyX(x,y);
 console.log("Final values (in calling function) = " + x + " " + y);
function modifyX(someNumber,someString) {
  console.log("Values passed into function = " + someNumber + " and " +
someString);
  someNumber = someNumber + 10;
  someString = "HumptyDumpty";
  console.log("Values passed into function have been modified to = " +
someNumber + " and " + someString);
```

EVEN IF THE FUNCTION MODIFIES THE functival such that the function of the functival such that the function is the function of the func

```
var x = 10;
 var y = "Vitthal";
 console.log("Initial values (in calling function) = " + x + " " + y);
 modifyX(x,y);
 console.log("Final values (in calling function) = " + x + " " + y);
function modifyX(someNumber,someString) {
  console.log("Values passed into function = " + someNumber + " and " +
someString);
  someNumber = someNumber + 10;
  someString = "HumptyDumpty";
  console.log("Values passed into function have been modified to = " +
someNumber + " and " + someString);
```

EVEN IF THE FUNCTION MODIFIES THE VYAKIABLE, 164 ORIGINAL IS UNCHANGED.

```
var y = "Vitth a Wo VARIABLES, ONE A NUMBER, THE console.log("Initial values (in calling OTHER IS A STRING." + y); modifyX(x,y); console.log("Final values (in calling function) = " + x + " " + y);
```

```
function modifyX(someNumber, someString) {
  console.log("Values passed into function = " + someNumber + " and " +
  someString);
  someNumber = someNumber + 10;
  someString = "HumptyDumpty";
  console.log("Values passed into function have been modified to = " +
  someNumber + " and " + someString);
}
```

EVEN IF THE FUNCTION MODIFIES THE funct WAR AND EP. FINE ORIGINAL IS UNCHANGED.

```
var y = "Vitthal";
console.log("Initial values (in calling function) = " + x + " " + y);
modifyX(x,y);
```

console.log("Final valuesTHENVARIABLESMAREMASSED INFO'A" + y);
FUNCTION..

```
function modifyX(someNumber,someString) {
  console.log("Values passed into function = " + someNumber + " and " +
  someString);
  someNumber = someNumber + 10;
  someString = "HumptyDumpty";
  console.log("Values passed into function have been modified to = " +
  someNumber + " and " + someString);
}
```

EVEN IF THE FUNCTION MODIFIES THE function was VARIABLE, THE ORIGINAL IS UNCHANGED.

```
console.log("Initial values (in calling function) = " + x + " " + y);
modifyX(x,y);
console.log("Final values (in calling function) = " + x + " " + y);
```

AND THAT FUNCTION MODIFIES THE VALUES OF THE VARIABLES PASSED IN function modifyX(someNumber, someString) {

console.log("Values passed into function = " + someNumber + " and " +
someString);

someNumber = someNumber + 10; someString = "HumptyDumpty";

```
console.log("Values passed into function have been modified to = " +
someNumber + " and " + someString);
```

EVEN IF THE FUNCTION MODIFIES THE functivarily functivarily functivarily for the functivarily functivarily for the functivarily for the functivarily functivarily for the functivarily functivarily for the function of the fu

```
var x = 10;
 var y = "Vitthal";
 console.log("Initial values (in calling function) = " + x + " " + y);
 modifyX(x,y);
 console.log("Final values (in calling function) = " + x + " " + y);
function modifyX(someNumber,someString) {
  console.log("Values passed into function = " + someNumber + " and " +
someString);
  someNumber = someNumber + 10;
  someString = "HumptyDumpty";
  console.log("Values passed into function have been modified to = " +
someNumber + " and " + someString);
```

EVEN IF THE FUNCTION MODIFIES THE TUNCTION WODIFIES THE ORIGINAL IS UNCHANGED.

```
Initial values (in calling function) = 10 Vitthal
 Values passed into function = 10 and Vitthal
 Values passed into function have been modified to = 20 and HumptyDumpty
Final values (in calling function) = 10 Vitthal
```

EVEN IF THE FUNCTION MODIFIES THE VARIABLE, THE ORIGINAL IS UNCHANGED.

Initial values (in calling function) = 10 Vitthal

Values passed into function = 10 and Vitthal

Values passed into function have been modified to = 20 and HumptyDumpty

Final values (in calling function) = 10 Vitthal

EVEN IF THE FUNCTION MODIFIES THE VARIABLE, THE ORIGINAL IS UNCHANGED.

Initial values (in calling function) = 10 Vitthal

Values passed into function = 10 and Vitthal

Values passed into function have been modified to = 20 and HumptyDumpty

Final values (in calling function) = 10 Vitthal

EVEN IF THE FUNCTION MODIFIES THE VARIABLE, THE ORIGINAL IS UNCHANGED.

Initial values (in calling function) = 10 Vitthal

Values passed into function = 10 and Vitthal

Values passed into function have been modified to = 20 and HumptyDumpty

Final values (in calling function) = 10 Vitthal