

1. Realizar una función que dada una fecha de nacimiento te devuelva el signo zodiacal 1

```
File Edit Selection View Go Run Terminal Help
Repaso1.ts - Dia3.EjerciciosRepas - Visual Studio Code

TS Repaso1.ts x
Repaso1 > TS Repaso1.ts > zodiaco
1 // Realizar una función que dada una fecha de
  nacimiento te devuelva el signo zodiacal
2
3 function zodiaco (dia:number, mes:number){
4   let signo = ""
5   switch(mes){
6     case 1):
7       if (dia >= 21) {
8         signo = "Acuario";
9       } else {
10        signo = "Capricornio";
11      }
12      break;
13     case 2:
14       if (dia >= 19) {
15         signo = "Piscis";
16       } else {
17         signo = "Acuario";
18       }
19       break;
20     case 3:
21       if (dia >= 20) {
22         signo = "Aries";
23       } else {
24         signo = "Piscis";
25       }
26       break;
27     case 4:
28       if (dia >= 20) {
29         signo = "Tauro";
30       } else {
31         signo = "Aries";
32       }
33       break;
34     case 5:
35       if (dia >= 21) {
36         signo = "Géminis";
37       } else {
38         signo = "Tauro";
39       }
40       break;
41     case 6:
42       if (dia >= 20) {
43         signo = "Cáncer";
44       } else {
45         signo = "Géminis";
46       }
47       break;
48     case 7:
49       if (dia >= 22) {
50         signo = "Leo";
51       } else {
52         signo = "Cáncer";
53       }
54       break;
55     case 8:
56       if (dia >= 21) {
57         signo = "Virgo";
58       } else {
59         signo = "Leo";
60       }
61       break;
62     case 9:
63       if (dia >= 22) {
64         signo = "Libra";
65       } else {
66         signo = "Virgo";
67       }
68       break;
69     case 10:
70       if (dia >= 22) {
71         signo = "Escorpión";
72       } else {
73         signo = "Libra";
74       }
75       break;
76     case 11:
77       if (dia >= 21) {
78         signo = "Sagitario";
79       } else {
80         signo = "Escorpión";
81       }
82       break;
83     case 12:
84       if (dia >= 21) {
85         signo = "Capricornio";
86       } else {
87         signo = "Sagitario";
88       }
89       break;
90   }
91   return "Tu signo del zodiaco es: " + signo;
92 }
93
94 console.log(zodiaco (25,6))
95
96 console.log(zodiaco(12, 4))
97
98 console.log(zodiaco (1,12))
99
100 console.log(zodiaco(31, 9))
101
102
```

Microsoft Windows [Versión 10.0.18362.900]
(c) 2019 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>tsc Repaso1

C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>node Repaso1

Tu signo del zodiaco es: Cáncer
Tu signo del zodiaco es: Aries
Tu signo del zodiaco es: Sagitario
Tu signo del zodiaco es: Libra

C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>

Ln 10, Col 39 Spaces: 4 UTF-8 CRLF TypeScript 3.9.4 17:19

2. Realizar un procedimiento que dado un país te imprima en que continente estás. (Max 5 países por continente)

The image shows a Visual Studio Code editor with a TypeScript file named `Repaso2.ts`. The file contains a function `continente` that takes a string `pais` and returns the continent it belongs to. The function uses a series of `if` and `else if` statements to map specific countries to their respective continents. The countries included are: España, Francia, Italia, Alemania, Portugal, China, Japón, Filipinas, Turkmenistán, Mongolia, Colombia, Panamá, Canadá, Estados Unidos, Jamaica, Luanda, Argel, Porto Novo, and El Cairo.

```
1 //Realizar un procedimiento que dado un país te imprima en que continente
  estás.(Max 5 países por continente)
2
3 function continente (pais:string){
4   let miContinente = ""
5   if (pais == "España"){
6     miContinente = "Europa";
7   }else if (pais == "Francia"){
8     miContinente = "Europa";
9   } else if (pais == "Italia") {
10    miContinente = "Europa";
11   } else if (pais == "Alemania"){
12    miContinente = "Europa";
13   } else if (pais == "Portugal") {
14    miContinente = "Europa";
15   }
16
17   if (pais == "China"){
18     miContinente = "Asia";
19   }else if (pais == "Japón"){
20     miContinente = "Asia";
21   } else if (pais == "Filipinas") {
22     miContinente = "Asia";
23   } else if (pais == "Turkmenistán"){
24     miContinente = "Asia";
25   } else if (pais == "Mongolia") {
26     miContinente = "Asia";
27   }
28
29   if (pais == "Colombia"){
30     miContinente = "América";
31   }else if (pais == "Panamá"){
32     miContinente = "América";
33   } else if (pais == "Canadá") {
34     miContinente = "América";
35   } else if (pais == "Estados Unidos"){
36     miContinente = "América";
37   } else if (pais == "Jamaica") {
38     miContinente = "América";
39   }
40
41   if (pais == "Luanda"){
42     miContinente = "África";
43   }else if (pais == "Argel"){
44     miContinente = "África";
45   } else if (pais == "Porto Novo") {
46     miContinente = "África";
47   } else if (pais == "El Cairo"){
48     miContinente = "África";
49   }
50
51   if (pais == "Luanda"){
52     miContinente = "África";
53   }else if (pais == "Argel"){
54     miContinente = "África";
55   } else if (pais == "Porto Novo") {
56     miContinente = "África";
57   } else if (pais == "El Cairo"){
58     miContinente = "África";
59   } else if (pais == "Gaborone") {
60     miContinente = "África";
61   }
62
63   if (pais == "Australia"){
64     miContinente = "Oceania";
65   }else if (pais == "Nueva Zelanda"){
66     miContinente = "Oceania";
67   } else if (pais == "Islas Marshall") {
68     miContinente = "Oceania";
69   } else if (pais == "Samoa"){
70     miContinente = "Oceania";
71   } else if (pais == "Papúa Nueva Guinea") {
72     miContinente = "Oceania";
73   }
74
75   return "El país introducido pertenece a " + miContinente
76 }
77
78 console.log(continente("Alemania"))
79 console.log(continente("Mongolia"))
80 console.log(continente("Panamá"))
81 console.log(continente("Porto Novo"))
82 console.log(continente("Samoa"))
```

The terminal output shows the results of the function calls:

```
El país introducido pertenece a
C:\Users\Usuario\Documents\Codenotch\Se
mana_3\Dia3.EjerciciosRepas>tsc Repaso2
Repaso2.ts:19:1 - error TS1128: Declara
tion or statement expected.
19 }
   *

Found 1 error.

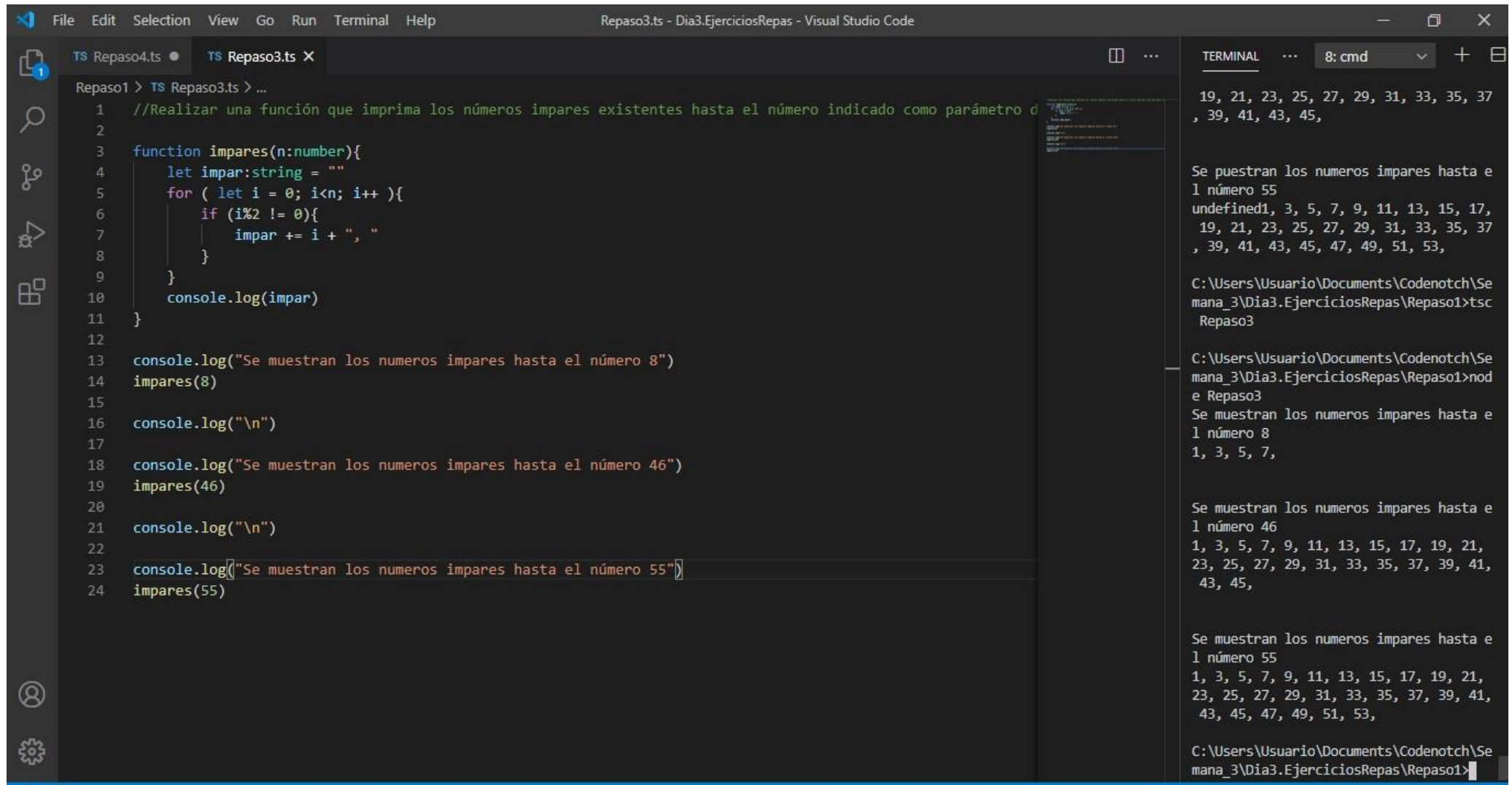
C:\Users\Usuario\Documents\Codenotch\Se
mana_3\Dia3.EjerciciosRepas>tsc Repaso2

C:\Users\Usuario\Documents\Codenotch\Se
mana_3\Dia3.EjerciciosRepas>node Repaso
2
El país introducido pertenece a Europa
C:\Users\Usuario\Documents\Codenotch\Se
mana_3\Dia3.EjerciciosRepas>tsc Repaso2

C:\Users\Usuario\Documents\Codenotch\Se
mana_3\Dia3.EjerciciosRepas>node Repaso
2
El país introducido pertenece a Europa
El país introducido pertenece a Asia
C:\Users\Usuario\Documents\Codenotch\Se
mana_3\Dia3.EjerciciosRepas>tsc Repaso2

C:\Users\Usuario\Documents\Codenotch\Se
mana_3\Dia3.EjerciciosRepas>node Repaso
2
El país introducido pertenece a Europa
El país introducido pertenece a Asia
El país introducido pertenece a América
El país introducido pertenece a África
El país introducido pertenece a Oceania
C:\Users\Usuario\Documents\Codenotch\Se
mana_3\Dia3.EjerciciosRepas>
```

3. Realizar una función que imprima los números impares existentes hasta el número indicado como parámetro de entrada.



The image shows a Visual Studio Code editor window with a file named `Repaso3.ts`. The code defines a function `impares` that takes a number `n` and prints all odd numbers from 1 to `n`. The function uses a `for` loop and a conditional `if (i % 2 !== 0)` to check for odd numbers. The function is called three times with arguments 8, 46, and 55. The terminal on the right shows the output of these calls, displaying the odd numbers for each range.

```
1 //Realizar una función que imprima los números impares existentes hasta el número indicado como parámetro de entrada
2
3 function impares(n:number){
4     let impar:string = ""
5     for ( let i = 0; i<n; i++ ){
6         if (i%2 !== 0){
7             impar += i + ", "
8         }
9     }
10    console.log(impar)
11 }
12
13 console.log("Se muestran los numeros impares hasta el número 8")
14 impares(8)
15
16 console.log("\n")
17
18 console.log("Se muestran los numeros impares hasta el número 46")
19 impares(46)
20
21 console.log("\n")
22
23 console.log("Se muestran los numeros impares hasta el número 55")
24 impares(55)
```

Terminal Output:

```
19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45,

Se muestran los numeros impares hasta el número 8
undefined1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53,

C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>tsc Repaso3

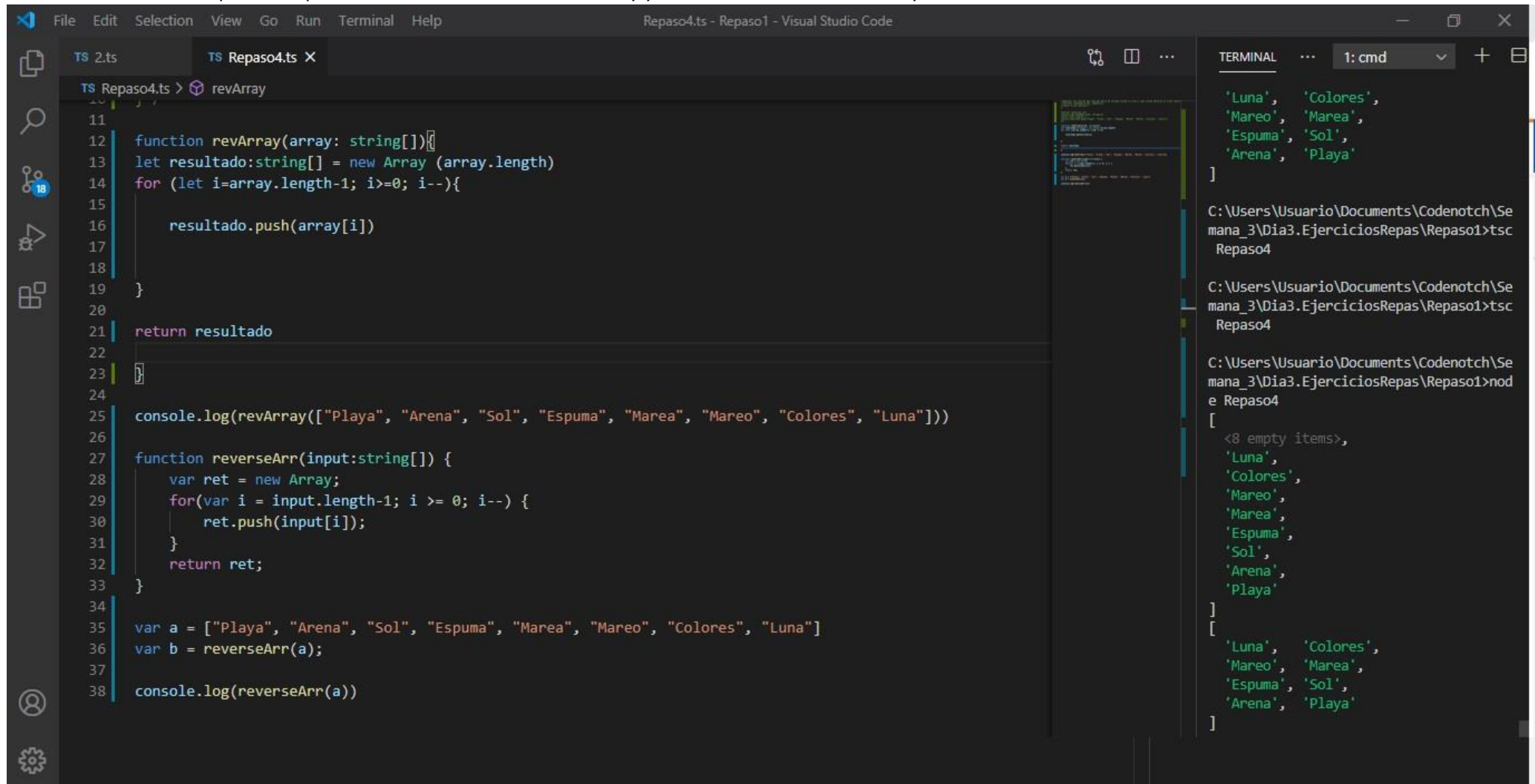
C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>node Repaso3
Se muestran los numeros impares hasta el número 8
1, 3, 5, 7,

Se muestran los numeros impares hasta el número 46
1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45,

Se muestran los numeros impares hasta el número 55
1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53,

C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>
```


4. Realizar una función que como parámetro de entrada reciba un array y como salida devuelva el array revertido.



The image shows a Visual Studio Code editor with a TypeScript file named `Repaso4.ts`. The code defines two functions: `revArray` and `reverseArr`, both of which reverse a string array. The `revArray` function uses a `for` loop to iterate from the end of the array to the beginning, pushing elements into a new array. The `reverseArr` function uses a `for` loop to iterate from the end of the array to the beginning, pushing elements into a new array. The code also includes a test case where an array `a` is defined with the elements `["Playa", "Arena", "Sol", "Espuma", "Marea", "Mareo", "Colores", "Luna"]`, and the `reverseArr` function is called on it, with the result logged to the console.

```
TS Repaso4.ts > revArray
11
12 function revArray(array: string[]) {
13   let resultado: string[] = new Array(array.length)
14   for (let i = array.length - 1; i >= 0; i--) {
15     resultado.push(array[i])
16   }
17   return resultado
18 }
19
20
21
22
23
24
25 console.log(revArray(["Playa", "Arena", "Sol", "Espuma", "Marea", "Mareo", "Colores", "Luna"]))
26
27 function reverseArr(input: string[]) {
28   var ret = new Array;
29   for (var i = input.length - 1; i >= 0; i--) {
30     ret.push(input[i]);
31   }
32   return ret;
33 }
34
35 var a = ["Playa", "Arena", "Sol", "Espuma", "Marea", "Mareo", "Colores", "Luna"]
36 var b = reverseArr(a);
37
38 console.log(reverseArr(a))
```

The terminal output shows the result of the `console.log` statement, which is an array of 8 empty items, followed by the reversed array elements: `'Luna', 'Colores', 'Mareo', 'Marea', 'Espuma', 'Sol', 'Arena', 'Playa'`.

```
C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>tsc Repaso4
C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>tsc Repaso4
C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>node Repaso4
[
  <8 empty items>,
  'Luna',
  'Colores',
  'Mareo',
  'Marea',
  'Espuma',
  'Sol',
  'Arena',
  'Playa'
]
```

5. Realizar una función que reciba como parámetro un array de strings que contenga colores y te imprima en cada caso si el color está en el arcoíris o no.

File Edit Selection View Go Run Terminal Help • Repaso5.ts - Dia3.EjerciciosRepas - Visual Studio Code

TS Repaso5.ts •

Repaso1 > TS Repaso5.ts > ...

```
1 //Realizar una función que reciba como parámetro un array de strings que contenga colores y te imprima en c
2
3 function arcoiris (colores:string[]){
4
5
6
7     if (colores.includes("red")){
8         console.log("El color rojo está en el arcoiris")
9     }
10    if (colores.includes("green")){
11        console.log("El color verde está en el arcoiris")
12    }
13    if (colores.includes("blue") ){
14        console.log("El color azul está en el arcoiris")
15    }
16
17 }
18
19 arcoiris(["green", "red", "orange", "blue", "purple"])
20
21 //Aunque me da error, está bien
22
23
24
25
```

TERMINAL ... 5: cmd +

Repaso5

Repaso5.ts:7:21 - error TS2339: Property 'includes' does not exist on type 'string[]'.

7 if (colores.includes("red")){

Repaso5.ts:10:21 - error TS2339: Property 'includes' does not exist on type 'string[]'.

10 if (colores.includes("green")){

Repaso5.ts:13:21 - error TS2339: Property 'includes' does not exist on type 'string[]'.

13 if (colores.includes("blue")){

Found 3 errors.

C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>node Repaso5

El color rojo está en el arcoiris

El color verde está en el arcoiris

El color azul está en el arcoiris

C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>

> OPEN EDITORS

TS Repaso5.ts > ...

55

56

57

58

59

60

61

62

63

64

65

66

67

68

69

70

71

72

73

74

75

76

77

78

79

80

81

82

83

84

85

else if (resultado.includes("orange")){

console.log("El color naranja está en el arcoiris")

else if (resultado.includes("purple")){

console.log("El color lila está en el arcoiris")

else if (resultado.includes("yellow")){

console.log("El color amarillo está en el arcoiris")

else {

console.log("Ese color no está en el arcoiris")

++

break;

while (i<colores.length)

function arcoiris (colores:string[]){

for(let i=0 ; i<colores.length; i++)

if (colores[i] == "red"||colores[i] == "blue"||colores[i]=="orange"||colores[i]=="purple"||colores[i] == "yellow"){

console.log("El color " + colores[i] + " está en el arcoiris")

else {

console.log("Ese color no está en el arcoiris")

6. Realizar una función que te devuelva si existe un número par en el array de números que introduces como parámetro de entrada.

The image shows a Visual Studio Code editor window with the file Repaso6.ts open. The Explorer sidebar on the left shows a project structure with folders 'DIA3.EJERCICIOSREPAS' and 'Repaso2', and various files including Repaso6.ts. The main editor area displays the following TypeScript code:

```
29 console.log(pares([24,435,257,76,355]))*/
30
31 function pares(numeros:number[]){
32
33     let rest: boolean
34
35     for ( let i = 0; i<numeros.length; i++ ){
36
37         if (numeros[i]%2 == 0)
38         {
39             rest= true
40         }else {
41             rest=false
42         }
43
44         return rest
45     }
46 }
47 console.log("El array [8,32,53,63,64,86,45] contiene números pares: ")
48 console.log(pares([8,32,53,63,64,86,45]))
49
50 console.log("\n")
51 console.log("El array [345,363,23,461] contiene números pares: ")
52 console.log(pares([345,363,23,461]))
53
54 console.log("\n")
55 console.log("El array [24,435,257,76,355] contiene números pares: ")
56 console.log(pares([24,435,257,76,355]))
```

The TERMINAL panel on the right shows the command prompt output:

```
C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>
C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>
C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>
C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>
C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>
C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>
C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>
C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>
C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>
C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>tsc Repaso6

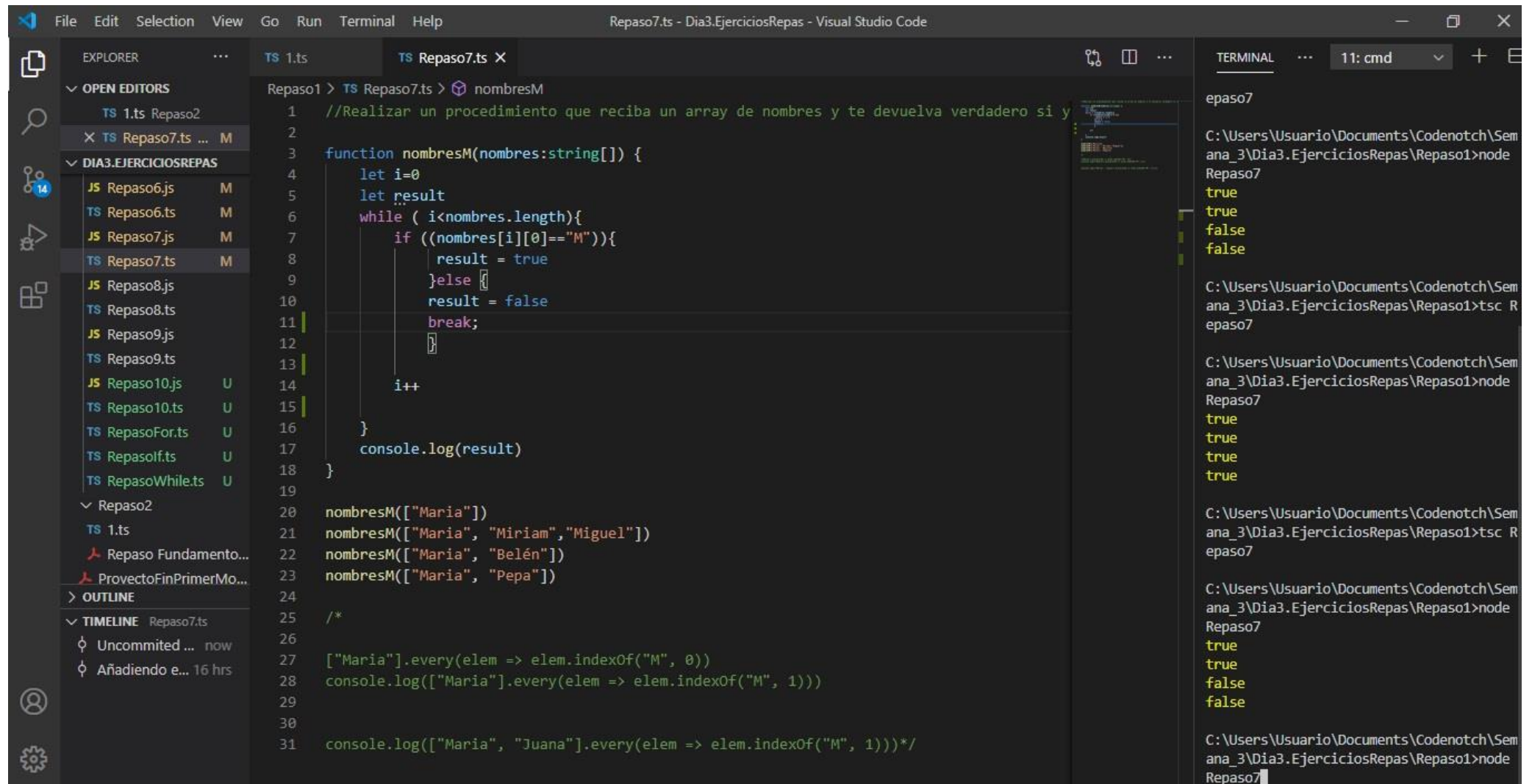
C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>node Repaso6
El array [8,32,53,63,64,86,45] contiene números pares:
true

El array [345,363,23,461] contiene números pares:
false

El array [24,435,257,76,355] contiene números pares:
true

C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>
```


7. Realizar un procedimiento que reciba un array de nombres y te devuelva verdadero si y solo si todos los nombres empiezan por M.



```
File Edit Selection View Go Run Terminal Help
Repaso7.ts - Dia3.EjerciciosRepas - Visual Studio Code

EXPLORER
OPEN EDITORS
  TS 1.ts Repaso2
  X TS Repaso7.ts ... M
DIA3.EJERCICIOSREPAS
  JS Repaso6.js M
  TS Repaso6.ts M
  JS Repaso7.js M
  TS Repaso7.ts M
  JS Repaso8.js
  TS Repaso8.ts
  JS Repaso9.js
  TS Repaso9.ts
  JS Repaso10.js U
  TS Repaso10.ts U
  TS RepasoFor.ts U
  TS RepasoIf.ts U
  TS RepasoWhile.ts U
  Repaso2
  TS 1.ts
  Repaso Fundamento...
  ProyectoFinPrimerMo...
OUTLINE
TIMELINE Repaso7.ts
  Uncommitted ... now
  Añadiendo e... 16 hrs

TS 1.ts
Repaso1 > TS Repaso7.ts > nombresM
1 //Realizar un procedimiento que reciba un array de nombres y te devuelva verdadero si y
2
3 function nombresM(nombres:string[]) {
4   let i=0
5   let result
6   while ( i<nombres.length){
7     if ((nombres[i][0]=="M")){
8       result = true
9     }else {
10      result = false
11     }
12     break;
13
14     i++
15
16   }
17   console.log(result)
18 }
19
20 nombresM(["Maria"])
21 nombresM(["Maria", "Miriam","Miguel"])
22 nombresM(["Maria", "Belén"])
23 nombresM(["Maria", "Pepa"])
24
25 /*
26
27 ["Maria"].every(elem => elem.indexOf("M", 0))
28 console.log(["Maria"].every(elem => elem.indexOf("M", 1)))
29
30
31 console.log(["Maria", "Juana"].every(elem => elem.indexOf("M", 1)))/
```

```
epaso7
C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>node Repaso7
true
true
false
false

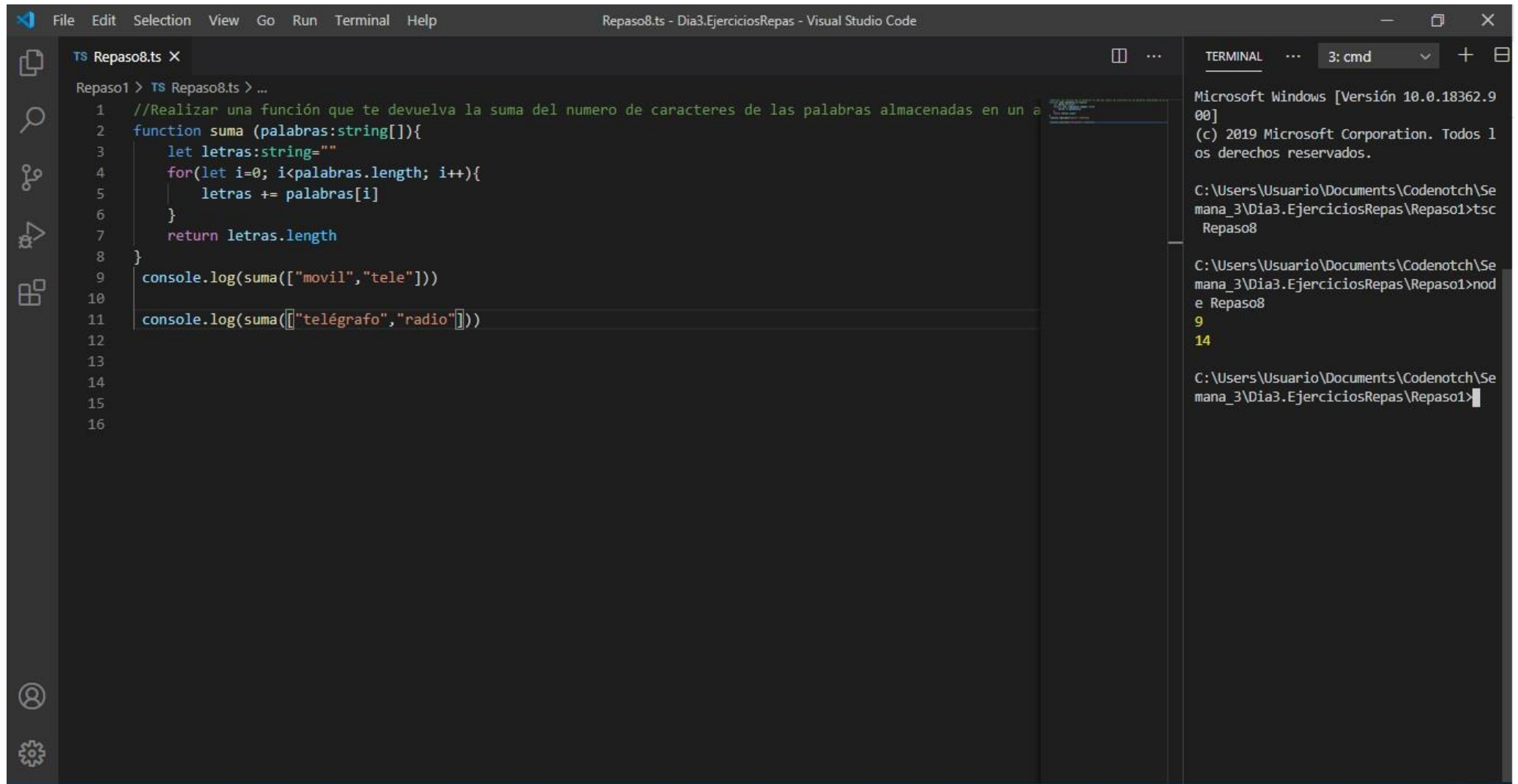
C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>tsc Repaso7
true
true
true
true

C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>node Repaso7
true
true
true
true

C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>tsc Repaso7
true
true
false
false

C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>node Repaso7
```

8. Realizar una función que te devuelva la suma del número de caracteres de las palabras almacenadas en un array.



The image shows a Visual Studio Code editor window with a file named `Repaso8.ts`. The code defines a function `suma` that takes an array of strings and returns the total number of characters. The function is tested with two arrays: `["movil", "tele"]` and `["telégrafo", "radio"]`. The terminal window on the right shows the command `node Repaso8` being executed, resulting in the output `9` and `14`.

```
TS Repaso8.ts X
Repaso1 > TS Repaso8.ts > ...
1 //Realizar una función que te devuelva la suma del numero de caracteres de las palabras almacenadas en un a
2 function suma (palabras:string[]){
3     let letras:string=""
4     for(let i=0; i<palabras.length; i++){
5         letras += palabras[i]
6     }
7     return letras.length
8 }
9 console.log(suma(["movil", "tele"]))
10
11 console.log(suma(["telégrafo", "radio"]))
12
13
14
15
16
```

Microsoft Windows [Versión 10.0.18362.900]
(c) 2019 Microsoft Corporation. Todos los derechos reservados.

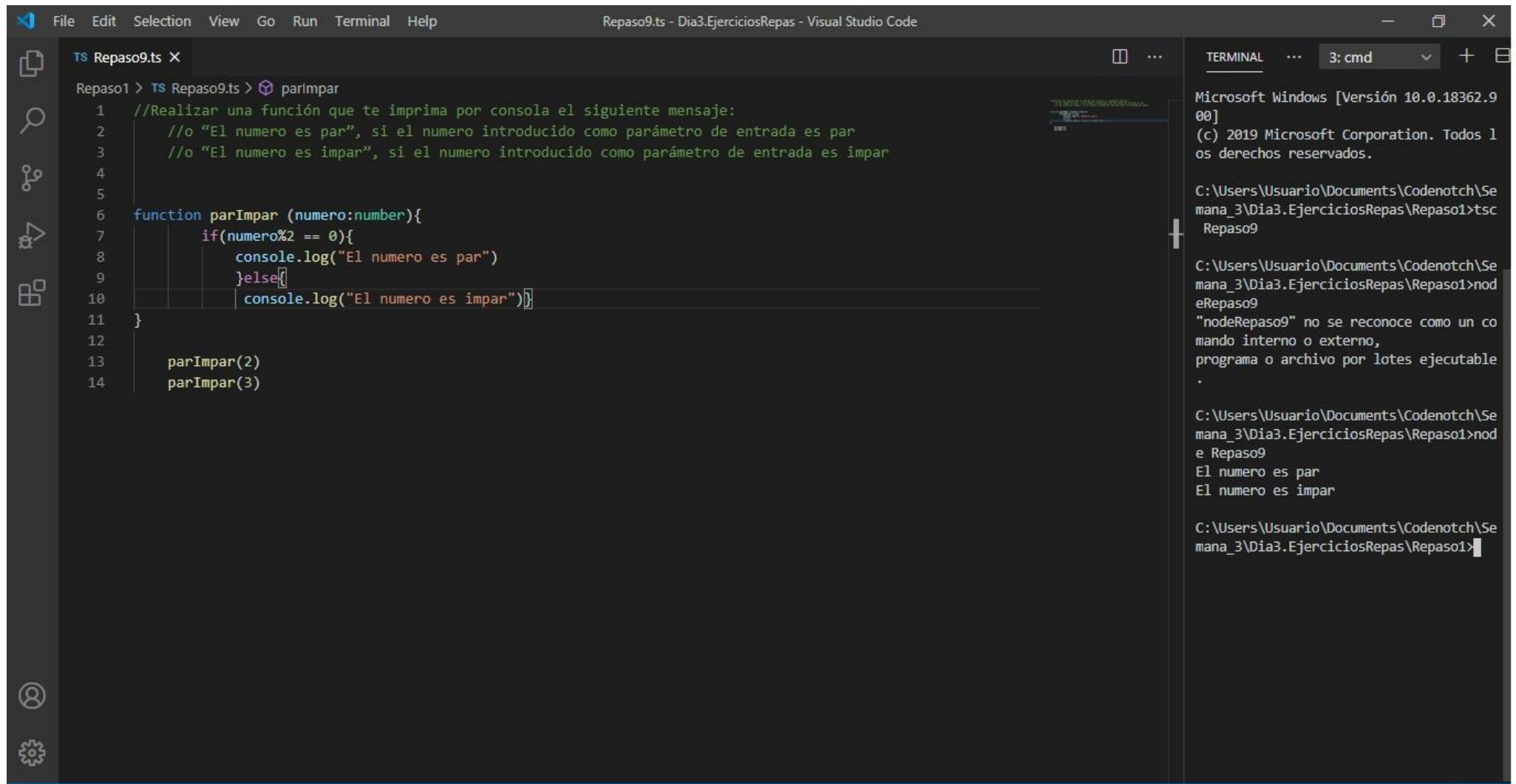
C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>tsc Repaso8

C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>node Repaso8

9
14

C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas\Repaso1>

9. Realizar una función que te imprima por consola el siguiente mensaje: o “El número es par”, si el numero introducido como parámetro de entrada es par o “El número es impar”, si el numero introducido como parámetro de entrada es impar



The image shows a Visual Studio Code editor window with a TypeScript file named `Repaso9.ts` and a terminal window open on the right.

TS Repaso9.ts

```
Repaso1 > TS Repaso9.ts > parImpar
1 //Realizar una función que te imprima por consola el siguiente mensaje:
2 //o "El numero es par", si el numero introducido como parámetro de entrada es par
3 //o "El numero es impar", si el numero introducido como parámetro de entrada es impar
4
5
6 function parImpar (numero:number){
7     if(numero%2 == 0){
8         console.log("El numero es par")
9     }else{
10        console.log("El numero es impar")
11    }
12
13    parImpar(2)
14    parImpar(3)
```

TERMINAL 3: cmd

```
Microsoft Windows [Versión 10.0.18362.900]
(c) 2019 Microsoft Corporation. Todos los derechos reservados.

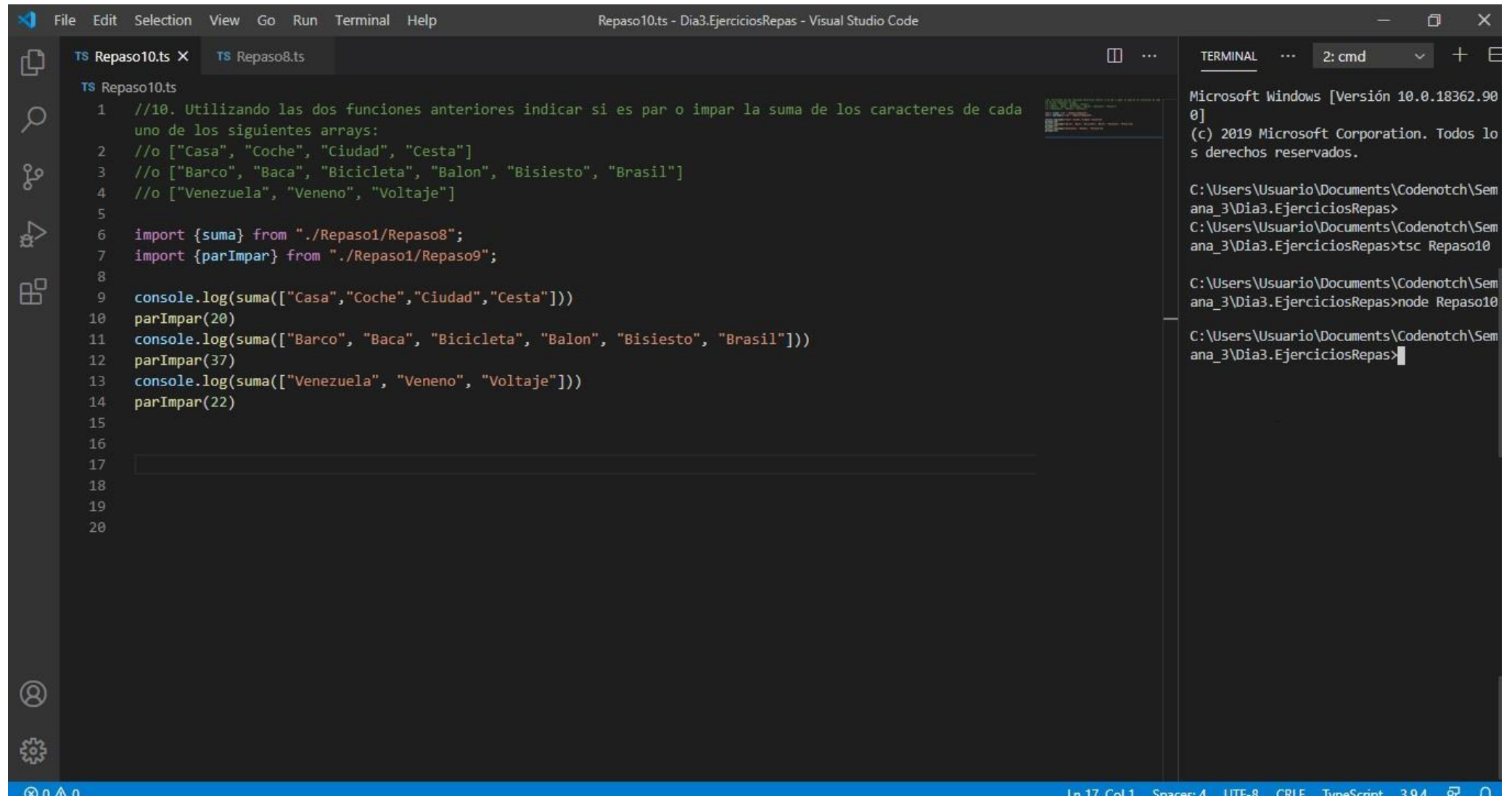
C:\Users\Usuario\Documents\Codenotch\Semana_3\EjerciciosRepas\Repaso1>tsc
Repaso9

C:\Users\Usuario\Documents\Codenotch\Semana_3\EjerciciosRepas\Repaso1>node
Repaso9
"nodeRepaso9" no se reconoce como un comando interno o externo,
programa o archivo por lotes ejecutable
.

C:\Users\Usuario\Documents\Codenotch\Semana_3\EjerciciosRepas\Repaso1>nod
e Repaso9
El numero es par
El numero es impar

C:\Users\Usuario\Documents\Codenotch\Semana_3\EjerciciosRepas\Repaso1>
```

10. Utilizando las dos funciones anteriores indicar si es par o impar la suma de los caracteres de cada uno de los siguientes arrays: o ["Casa", "Coche", "Ciudad", "Cesta"] o ["Barco", "Baca", "Bicicleta", "Balon", "Bisiesto", "Brasil"] o ["Venezuela", "Veneno", "Voltaje"]



The image shows a screenshot of the Visual Studio Code editor. The main editor window displays a TypeScript file named `Repaso10.ts`. The code in the file is as follows:

```
1 //10. Utilizando las dos funciones anteriores indicar si es par o impar la suma de los caracteres de cada
  uno de los siguientes arrays:
2 //o ["Casa", "Coche", "Ciudad", "Cesta"]
3 //o ["Barco", "Baca", "Bicicleta", "Balon", "Bisiesto", "Brasil"]
4 //o ["Venezuela", "Veneno", "Voltaje"]
5
6 import {suma} from "../Repaso1/Repaso8";
7 import {parImpar} from "../Repaso1/Repaso9";
8
9 console.log(suma(["Casa","Coche","Ciudad","Cesta"]))
10 parImpar(20)
11 console.log(suma(["Barco", "Baca", "Bicicleta", "Balon", "Bisiesto", "Brasil"]))
12 parImpar(37)
13 console.log(suma(["Venezuela", "Veneno", "Voltaje"]))
14 parImpar(22)
15
16
17
18
19
20
```

The right sidebar shows the `TERMINAL` panel with the command prompt. The terminal output shows the execution of the command `tsc Repaso10` and the resulting command prompt `C:\Users\Usuario\Documents\Codenotch\Semana_3\Dia3.EjerciciosRepas>`.

12. Crear un proyecto nuevo en Git que se llame RepasoProgramacion y guardar los otros ficheros allí

The image shows a terminal window on the left and a GitHub repository page on the right.

Terminal Window:

```
create mode 100644 node_modules/@types/node/ts3.5/base.d.ts
create mode 100644 node_modules/@types/node/ts3.5/global.d.ts
create mode 100644 node_modules/@types/node/ts3.5/index.d.ts
create mode 100644 node_modules/@types/node/ts3.5/ssl.d.ts
create mode 100644 node_modules/@types/node/ts3.7/assert.d.ts
create mode 100644 node_modules/@types/node/ts3.7/base.d.ts
create mode 100644 node_modules/@types/node/ts3.7/index.d.ts
create mode 100644 node_modules/@types/node/tty.d.ts
create mode 100644 node_modules/@types/node/util.d.ts
create mode 100644 node_modules/@types/node/v8.d.ts
create mode 100644 node_modules/@types/node/worker_threads.d.ts
create mode 100644 node_modules/@types/node/zlib.d.ts
create mode 100644 package-lock.json

User@DESKTOP-3068GJH MINGW64 ~/Documents/Codenotch/Semana_3/Dia3.EjerciciosRepas/Repaso1 (master)
$ git push
Enumerating objects: 99, done.
Counting objects: 100% (99/99), done.
Delta compression using up to 4 threads
Compressing objects: 100% (94/94), done.
Writing objects: 100% (98/98), 1.97 MiB | 413.00 KiB/s, done.
Total 98 (delta 4), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (4/4), done.
To https://github.com/ireneJP/Reto12Hoja1.git
 ae61fbc..3df1832 master -> master

User@DESKTOP-3068GJH MINGW64 ~/Documents/Codenotch/Semana_3/Dia3.EjerciciosRepas/Repaso1 (master)
$
```

GitHub Repository Page:

The page shows the repository `ireneJP/Reto12Hoja1` with 2 commits and 1 branch. The commit history is as follows:

File	Commit Message	Time
node_modules/@types/node	Añadiendo ejercicios hoja1	39 seconds ago
1.jpg	Añadiendo ejercicios hoja1	39 seconds ago
10.jpg	Añadiendo ejercicios hoja1	39 seconds ago
2.jpg	Añadiendo ejercicios hoja1	39 seconds ago
3.jpg	Añadiendo ejercicios hoja1	39 seconds ago
4.jpg	Añadiendo ejercicios hoja1	39 seconds ago
5.jpg	Añadiendo ejercicios hoja1	39 seconds ago
6.jpg	Añadiendo ejercicios hoja1	39 seconds ago
7.jpg	Añadiendo ejercicios hoja1	39 seconds ago
8.jpg	Añadiendo ejercicios hoja1	39 seconds ago
9.jpg	Añadiendo ejercicios hoja1	39 seconds ago
README.md	first commit	9 minutes ago
Repaso Fundamentos1.pdf	Añadiendo ejercicios hoja1	39 seconds ago
Repaso1.js	Añadiendo ejercicios hoja1	39 seconds ago
Repaso1.ts	Añadiendo ejercicios hoja1	39 seconds ago
Repaso2.js	Añadiendo ejercicios hoja1	39 seconds ago
Repaso2.ts	Añadiendo ejercicios hoja1	39 seconds ago
Repaso3.js	Añadiendo ejercicios hoja1	39 seconds ago
Repaso3.ts	Añadiendo ejercicios hoja1	39 seconds ago
Repaso4.ts	Añadiendo ejercicios hoja1	39 seconds ago
Repaso5.js	Añadiendo ejercicios hoja1	39 seconds ago

The repository also includes a README section, a Releases section (No releases published), a Packages section (No packages published), and a Languages section (TypeScript 32.5%, JavaScript 67.5%).