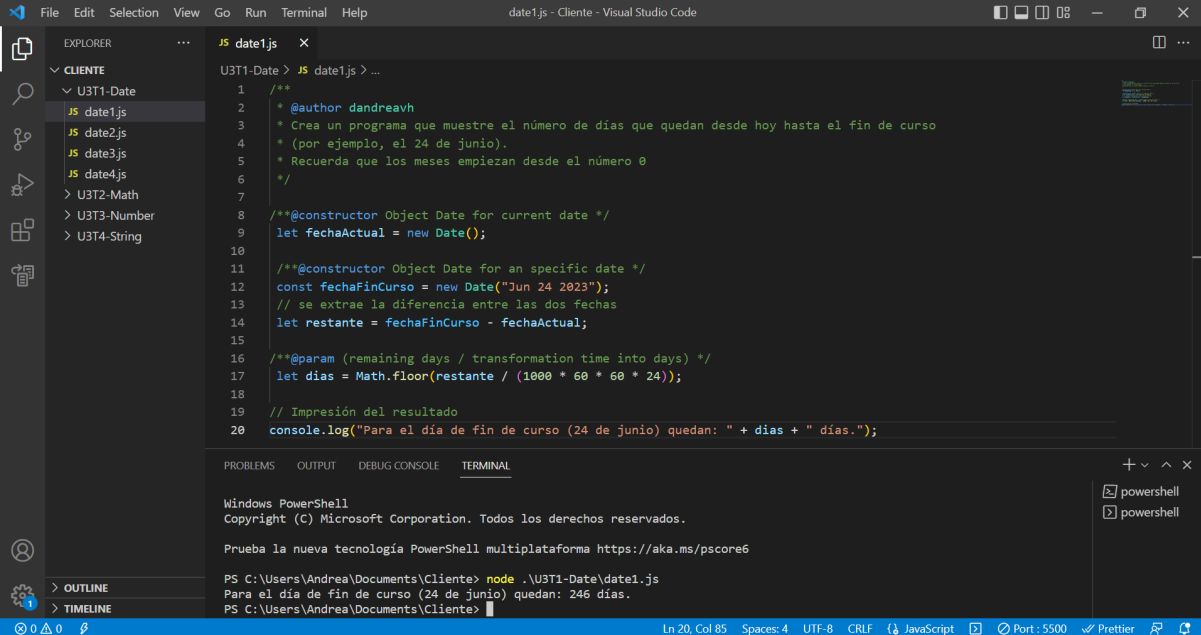


EJ1



The screenshot shows the Visual Studio Code interface. On the left, the Explorer pane shows a project structure with a folder named 'CLIENTE' containing subfolders 'U3T1-Date', 'U3T2-Math', 'U3T3-Number', and 'U3T4-String'. The 'U3T1-Date' folder is expanded, showing files 'date1.js', 'date2.js', 'date3.js', and 'date4.js'. The main editor displays the content of 'date1.js'. The file contains a JavaScript program that calculates the number of days remaining until June 24, 2023, from the current date. The program uses the Date object, Math.floor, and console.log. The terminal at the bottom shows the command 'node .\U3T1-Date\date1.js' being executed, resulting in the output 'Para el día de fin de curso (24 de junio) quedan: 246 días.'.

```
1  /**
2  * @author dandreavh
3  * Crea un programa que muestre el número de días que quedan desde hoy hasta el fin de curso
4  * (por ejemplo, el 24 de junio).
5  * Recuerda que los meses empiezan desde el número 0
6  */
7
8  /**@constructor Object Date for current date */
9  let fechaActual = new Date();
10
11 /**@constructor Object Date for an specific date */
12 const fechaFinCurso = new Date("Jun 24 2023");
13 // se extrae la diferencia entre las dos fechas
14 let restante = fechaFinCurso - fechaActual;
15
16 /**@param (remaining days / transformation time into days) */
17 let dias = Math.floor(restante / (1000 * 60 * 60 * 24));
18
19 // Impresión del resultado
20 console.log("Para el día de fin de curso (24 de junio) quedan: " + dias + " días.");
```

Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnología PowerShell multiplataforma <https://aka.ms/pscore6>

PS C:\Users\Andrea\Documents\Cliente> node .\U3T1-Date\date1.js
Para el día de fin de curso (24 de junio) quedan: 246 días.
PS C:\Users\Andrea\Documents\Cliente>

```
/**
 * @author dandreavh
 * Crea un programa que muestre el número de días que quedan desde hoy
hasta el fin de curso
 * (por ejemplo, el 24 de junio).
 * Recuerda que los meses empiezan desde el número 0
 */

/**@constructor Object Date for current date */
let fechaActual = new Date();

/**@constructor Object Date for an specific date */
const fechaFinCurso = new Date("Jun 24 2023");
// se extrae la diferencia entre las dos fechas
let restante = fechaFinCurso - fechaActual;

/**@param (remaining days / transformation time into days) */
let dias = Math.floor(restante / (1000 * 60 * 60 * 24));

// Impresión del resultado
console.log("Para el día de fin de curso (24 de junio) quedan: " + dias
+ " días.");
```

EJ2

```
1  /**
2  * @author dandreavh
3  * Crea un programa que pida por parámetro tu cumpleaños (no hace falta el año) y
4  * saque todos los años en que tu cumpleaños va a caer en domingo desde este año hasta el año 2100.
5  * Recuerda que los meses empiezan desde el número 0.
6  */
7  // Permite capturar un parámetro introducido de un array de elementos
8  const arguments = process.argv.slice(2);
9
10 // De la currentDate (que es el primer elemento del array anterior) que se pasa con este formato: dd/mm/aaaa, se tran
11 let separatedDate = arguments[0].split("/");
12
13 // Instancia de un objeto Date
14 let birthday = new Date();
15
16 // Se cambia el día, que se captura del anterior array, en la primera posición (necesita un incremento)
17 birthday.setDate(separatedDate[0]+1);
18
19 // Se cambia el mes, que se captura del anterior array, en la segunda posición (necesita el decremento)
20 birthday.setMonth(separatedDate[1]-1);
21
22 // Se captura el día
23 const day = birthday.getDate();
24
25 // Se captura el mes
26 const month = birthday.getMonth();
```

```
/**
 * @author dandreavh
 * Crea un programa que pida por parámetro tu cumpleaños (no hace falta
el año) y
 * saque todos los años en que tu cumpleaños va a caer en domingo desde
este año hasta el año 2100.
 * Recuerda que los meses empiezan desde el número 0.
 */
// Permite capturar un parámetro introducido de un array de elementos
const arguments = process.argv.slice(2);

// De la currentDate (que es el primer elemento del array anterior) que
se pasa con este formato: dd/mm/aaaa, se transforma en un array de 3
elementos
let separatedDate = arguments[0].split("/");

// Instancia de un objeto Date
let birthday = new Date();

// Se cambia el día, que se captura del anterior array, en la primera
posición (necesita un incremento)
birthday.setDate(separatedDate[0]+1);

// Se cambia el mes, que se captura del anterior array, en la segunda
posición (necesita el decremento)
```

```
birthday.setMonth(separatedDate[1]-1);

// Se captura el día
const day = birthday.getDate();

// Se captura el mes
const month = birthday.getMonth();

// Se pone por defecto el año
let year = 2022;

// Se declara el contador de las veces que ocurrirá que caiga en
domingo
let counter = 0;

// se ejecutará hasta que llegue a la fecha límite (2100)
while (birthday.getFullYear() != 2100) {
    // cuando coincida que el día sea 0 = domingo, el primer día de la
semana
    if (birthday.getDay() === 0) {
        // se incrementa el valor del contador
        counter++;
    }
    // independiente de la condición, se incrementa el valor del año
(punto de control del bucle)
    year++;
    // se actualiza el valor del año en la fecha del cumpleaños
    birthday.setFullYear(year);
}

// Impresión por consola del valor
console.log("Desde ahora, su cumpleaños caerá en domingo " + counter +
" veces");
```

EJ3

```
1  * @author dandreavh
2
3  * Crea un programa que muestre la fecha actual en diferentes formatos,
4  * según el valor que introduzca el usuario por parámetro:
5  *     15/10/2020
6  *     Jueves, 15 de octubre de 2020.
7  *     Thursday, October 15, 2020.
8  */
9
10 // Permite capturar un parámetro introducido de un array de elementos
11 const arguments = process.argv.slice(2);
12 // Captura de la elección del usuario del array anterior
13 const choise = arguments[0];
14 let date = new Date();
15 // Del objeto creado se obtienen los valores de interés: día, mes, año y día de la semana
16 let day = date.getDate();
17 let month = date.getMonth()+1;
18 let year = date.getFullYear();
19 let weekDay = date.getDay();
20
21 // Evaluación de la opción para la ejecución de las distintas estructuras
22 if(choise==1){
23     console.log("op1: "+day+"/"+month+"/"+year);
24 }else if(choise==2){
25     switch(weekDay){
26         case 0: weekDay = "Domingo"; break;
27         case 1: weekDay = "Lunes"; break;
```

PS C:\Users\Andrea\Documents\Cliente> node .\U3T1-Date\date3.js 2
op2: Jueves, 20 de octubre de 2022.
PS C:\Users\Andrea\Documents\Cliente>

```
/**
 * @author dandreavh
 *
 * Crea un programa que muestre la fecha actual en diferentes formatos,
 * según el valor que introduzca el usuario por parámetro:
 *
 *     15/10/2020
 *
 *     Jueves, 15 de octubre de 2020.
 *
 *     Thursday, October 15, 2020.
 */
// Permite capturar un parámetro introducido de un array de elementos
const arguments = process.argv.slice(2);
// Captura de la elección del usuario del array anterior
const choise = arguments[0];
let date = new Date();
// Del objeto creado se obtienen los valores de interés: día, mes, año
y día de la semana
let day = date.getDate();
let month = date.getMonth()+1;
let year = date.getFullYear();
let weekDay = date.getDay();

// Evaluación de la opción para la ejecución de las distintas
estructuras
if(choise==1){
    console.log("op1: "+day+"/"+month+"/"+year);
}else if(choise==2){
```

```
switch(weekDay) {
    case 0: weekDay = "Domingo"; break;
    case 1: weekDay = "Lunes"; break;
    case 2: weekDay = "Martes"; break;
    case 3: weekDay = "Miércoles"; break;
    case 4: weekDay = "Jueves"; break;
    case 5: weekDay = "Viernes"; break;
    case 6: weekDay = "Sábado"; break;
}
switch(month) {
    case 1: month = "enero"; break;
    case 2: month = "febrero"; break;
    case 3: month = "marzo"; break;
    case 4: month = "abril"; break;
    case 5: month = "mayo"; break;
    case 6: month = "junio"; break;
    case 7: month = "julio"; break;
    case 8: month = "agosto"; break;
    case 9: month = "septiembre"; break;
    case 10: month = "octubre"; break;
    case 11: month = "noviembre"; break;
    case 12: month = "diciembre"; break;
}
console.log("op2: "+weekDay+", "+day+" de "+month+" de "+year+".");

}else if(choise==3){
    // PREGUNTAR SI ES ESTE FORMATO O LITERALMENTE EL OTRO
    console.log("op3: " + date.toString());
}

}else{
    // Control de error del dato introducido como parámetro
    console.error("Wrong choise");
}
```

EJ4

```
1  /**
2   * @author dandreavh
3   * Crea un programa que muestre la hora actual en diferentes formatos,
4   * según el valor que introduzca el usuario por parámetro:
5   *     14:35:07 (hora detallada con minutos y segundos)
6   *     02:35 PM o 02:35:07 AM (hora con minutos y AM o PM según sea antes o después del medio día)
7   */
8   // Permite capturar un parámetro introducido de un array de elementos
9   const arguments = process.argv.slice(2);
10  // Captura de la elección del usuario del array anterior
11  const choice = arguments[0];
12  let date = new Date();
13  // Del objeto creado se obtienen los valores de interés: día, mes, año y día de la semana
14  let hour = date.getHours();
15  let minutes = date.getMinutes();
16  let seconds = date.getSeconds();
17
18  // Evaluación de la opción para la ejecución de las distintas estructuras
19  if(choice==1){
20      console.log("op1: "+hour+":"+minutes+": "+seconds);
21  }else if(choice==2){
22      // Comprobación AM o PM
23      let format = hour >= 12 ? 'PM' : 'AM';
24      // Cálculo a 12 horas
25      hour = hour%12;
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

PS C:\Users\Andrea\Documents\Cliente> node .\U3T1-Date\date4.js 2
op2: 8:33 PM
PS C:\Users\Andrea\Documents\Cliente>

```
/**
 * @author dandreavh
 * Crea un programa que muestre la hora actual en diferentes formatos,
 * según el valor que introduzca el usuario por parámetro:
 *     14:35:07 (hora detallada con minutos y segundos)
 *     02:35 PM o 02:35:07 AM (hora con minutos y AM o PM según sea
antes o después del medio día)
 */
// Permite capturar un parámetro introducido de un array de elementos
const arguments = process.argv.slice(2);
// Captura de la elección del usuario del array anterior
const choice = arguments[0];
let date = new Date();
// Del objeto creado se obtienen los valores de interés: día, mes, año
y día de la semana
let hour = date.getHours();
let minutes = date.getMinutes();
let seconds = date.getSeconds();

// Evaluación de la opción para la ejecución de las distintas
estructuras
if(choice==1){
    console.log("op1: "+hour+":"+minutes+": "+seconds);
}else if(choice==2){
    // Comprobación AM o PM
```

```
    let format = hour >= 12 ? 'PM' : 'AM';  
    // Cálculo a 12horas  
    hour = hour%12;  
    // Cambio a 12horas  
    hour = hour ? hour : 12;  
    minutes = minutes < 10 ? '0' + minutes : minutes;  
    console.log("op2: "+hour+": "+minutes+" "+format);  
  }else{  
    // Control de error del dato introducido como parámetro  
    console.error("Wrong choise");  
  }  
}
```