

## Работа с датой и временем в C# (DateTime)

Работа с датой и временем в C#, в основном, проводится с использованием структуры DateTime.

Для создания объекта типа DateTime можно воспользоваться одним из конструкторов структуры:

```
1. public DateTime()
2. public DateTime(int year, int month, int day)
3. public DateTime(int year, int month, int day, Calendar calendar)
4. public DateTime(int year, int month, int day, int hour, int minute, int second)
5. public DateTime(int year, int month, int day, int hour, int minute, int second, DateTimeKind kind)
6. public DateTime(int year, int month, int day, int hour, int minute, int second, Calendar calendar)
7. public DateTime(int year, int month, int day, int hour, int minute, int second, int millisecond)
8. public DateTime(int year, int month, int day, int hour, int minute, int second, int millisecond, DateTimeKind kind)
9. public DateTime(int year, int month, int day, int hour, int minute, int second, int millisecond, Calendar calendar)
10. public DateTime(int year, int month, int day, int hour, int minute, int second, int millisecond, Calendar calendar,
    DateTimeKind kind)
11. public DateTime(long ticks)
12. public DateTime(long ticks, DateTimeKind kind)
```

Пример 1:

```
Ссылка: 0
static void Main(string[] args)
{
    DateTime empty = new DateTime();
    DateTime date = new DateTime(2021, 09, 11);
    DateTime dateTime = new DateTime(2021, 09, 11, 10, 59, 59);
    Console.WriteLine(empty);
    Console.WriteLine(date);
    Console.WriteLine(dateTime);
}
```

В результате получим следующий вывод консоли:

```
01.01.0001 0:00:00
11.09.2021 0:00:00
11.09.2021 10:59:59
```

В первом случае мы использовали конструктор по умолчанию и получили минимально возможную дату и время, во втором случае мы задали только дату и в третьем — дату и время.

При этом, если необходимо получить текущие значения даты и времени в C#, то можно воспользоваться следующими свойствами DateTime:

```
Ссылка: 0
static void Main(string[] args)
{
    DateTime now = DateTime.Now; //текущие дата и время
    DateTime nowUtc = DateTime.UtcNow; //текущие дата и время в UTC
    DateTime dateUtc = DateTime.Today; //текущая дата
    Console.WriteLine($"Now: {now}");
    Console.WriteLine($"nowUtc: {nowUtc}");
    Console.WriteLine($"Today {dateUtc}");
    Console.ReadKey();
}
```

```
Now: 26.11.2023 23:21:13
nowUtc: 26.11.2023 17:21:13
Today 26.11.2023 0:00:00
```

## Работа с DateTime в C#

Структура DateTime содержит также ряд полезных свойств и методов, позволяющих манипулировать значениями даты и времени в своих приложениях.

Для того, чтобы получить из объекта типа DateTime только дату или только время, необходимо воспользоваться его свойствами Date или TimeOfDay:

```
Ссылка: 0
static void Main(string[] args)
{
    DateTime now = DateTime.Now; //текущие дата и время
    Console.WriteLine($"Дата и время: {now}");
    Console.WriteLine($"Дата: {now.Date}");
    Console.WriteLine($"Время: {now.TimeOfDay}");
    Console.ReadKey();
}
```

```
Дата и время: 26.11.2023 23:24:00
Дата: 26.11.2023 0:00:00
Время: 23:24:00.5399932
```

Обратите внимание, что время выводится вплоть до миллисекунд.

Как в C# получить из DateTime порядковый номер дня в году?

Чтобы получить порядковый номер дня в году, необходимо использовать свойство DayOfYear:

```
Ссылка: 0
static void Main(string[] args)
{
    DateTime now = new DateTime(2023, 09, 13);
    int day = now.DayOfYear;
    if (day == 256)
        Console.WriteLine($"День: {day}. Сегодня День программиста!");
    Console.ReadKey();
}
```

```
День: 256. Сегодня День программиста!
```

Формат	Результат
<code>DateTime.Now.ToString("D")</code>	22 ноября 2016 г.
<code>DateTime.Now.ToString("d")</code>	22.11.2016
<code>DateTime.Now.ToString("F")</code>	22 ноября 2016 г. 17:19:59
<code>DateTime.Now.ToString("f")</code>	22 ноября 2016 г. 17:19
<code>DateTime.Now.ToString("G")</code>	22.11.2016 17:19:59
<code>DateTime.Now.ToString("g")</code>	22.11.2016 17:19
<code>DateTime.Now.ToString("M")</code>	ноября 22

Формат	Результат
<i>DateTime.Now.ToString(«O»)</i>	2016-11-22T17:19:59.2628195+03:00
<i>DateTime.Now.ToString(«R»)</i>	Tue 22 Nov 2016 17:19:59 GMT
<i>DateTime.Now.ToString(«s»)</i>	2016-11-22T17:19:59
<i>DateTime.Now.ToString(«T»)</i>	17:19:59
<i>DateTime.Now.ToString(«t»)</i>	17:19
<i>DateTime.Now.ToString(«U»)</i>	22 ноября 2016 г. 14:19:59
<i>DateTime.Now.ToString(«u»)</i>	2016-11-22 17:19:59Z
<i>DateTime.Now.ToString(«Y»)</i>	Ноябрь 2016

Формат	Описание
d	День месяца. От 1 до 31 (без ведущего нуля)
dd	День месяца. От 01 до 31 (с ведущим нулем).
ddd	Краткое название дня недели
dddd	Полное название дня недели
M	Месяц. От 1 до 12 (без ведущего нуля)
MM	Месяц. От 01 до 12 (с ведущим нулем)
MMM	Краткое название месяца
MMMM	Полное название месяца
y	Одна или две последние цифры года (практически неактуально)
yy	Две последние цифры года
yyy	Три последние цифры года
yyyy	Четыре последние цифры года
yyyyy	Пять последних цифр года (будет не скоро)
h	Час. От 0 до 12. (без ведущего нуля)
hh	Час. От 00 до 12 (с ведущим нулем)
H	Час. От 0 до 23 (без ведущего нуля)
HH	Час. От 00 до 23 (с ведущим нулем)
m	Минута. От 0 до 59 (без ведущего нуля)
mm	Минута. От 00 до 59 (с ведущим нулем)
s	Секунда. От 0 до 59 (без ведущего нуля)
ss	Секунда. От 00 до 59 (с ведущим нулем)
от f до fffffff	Миллисекунды. Выводимое количество цифр соответствует количеству «f»
K	Часовой пояс
t	Половина дня. А или Р
tt	Половина дня. АМ или РМ
z	Смещение в часах относительно GMT (без ведущего нуля)
zz	Смещение в часах относительно GMT (с ведущим нулем)
g	Период или эра

## Самостоятельные работы:

1. **Определение текущей даты и времени.** Получите текущую дату и время в C#.
2. **Извлечение компонентов даты и времени.** Разберите текущую дату на составляющие (год, месяц, день) и выведите их.
3. **Форматирование даты.** Выведите текущую дату в различных форматах (короткий, длинный, пользовательский).
4. **Добавление времени.** Добавьте к текущей дате определенное количество часов, минут или секунд.
5. **Разница между датами.** Найдите разницу между двумя датами в днях, часах, минутах и секундах.
6. **Проверка на високосный год.** Определите, является ли текущий год високосным.
7. **Сравнение дат.** Сравните две даты и определите, какая из них раньше или позже.
8. **Определение дня недели.** Определите текущий день недели и выведите его название.
9. **Создание даты с пользовательскими значениями.** Позвольте пользователю ввести год, месяц и день, а затем создайте дату.
10. **Определение количества дней в месяце.** Выведите количество дней в текущем месяце.
11. **Определение, является ли дата будущей или прошлой.** Проверьте, является ли введенная пользователем дата будущей или прошлой относительно текущей.
12. **Переход на следующий/предыдущий день.** Создайте дату для следующего и предыдущего дней относительно текущей даты.
13. **Работа с таймзонами.** Выведите текущую дату и время в другой временной зоне.
14. **Подсчет количества рабочих дней между датами.** Определите количество рабочих дней между двумя датами, исключая выходные.
15. **Определение времени суток.** Выведите, является ли текущее время утром, днем, вечером или ночью.
16. **Проверка на наступление события в будущем.** Проверьте, наступит ли определенное событие (например, день рождения) в будущем или уже произошло.
17. **Вычисление возраста.** По дате рождения определите текущий возраст пользователя.