


# C#. Programowanie

## Jesse Liberty

Drogi Czytelniku! Poniżej zamieszczona jest errata do książki:
**"C#. Programowanie"**

Jest to lista błędów znalezionych po opublikowaniu książki, zgłoszonych i zaakceptowanych przez naszą redakcję. Pragniemy, aby nasze publikacje były wiarygodne i spełniały Twoje oczekiwania. Zapoznaj się z poniższą listą. Jeśli masz dodatkowe zastrzeżenia, możesz je zgłosić pod adresem https://ebookpoint.pl/user/erraty

| Strona | Linia | Jest | Powinno |
|---|---|---|---|
| 33 | 6-7 | można na przykład zmienić nazwę pliku Program.cs (znajduje się ona w oknie Solution Explorer) na hello.cs a nazwę klasy Program na HellowWorld. | można na przykład zmienić nazwę klasy Program na HellowWorld a nazwę pliku Program.cs (znajduje się ona w oknie Solution Explorer) na hello.cs. |
| 58 | 27 | W tym przypadku warunek 1 powoduje wykonanie instrukcji i można go "przepuścić" | W tym przypadku warunek 1 powoduje wykonanie instrukcji i nie można go "przepuścić" |
| 58 | 36 | Left na listingu 3.8: | Left na listingu 3.9: |
| 61 | 27 | return 0; | return; |
| 61 | 7 | do instrukcja while wyrazenie | do instrukcja while (wyrazenie) |
| 75 | 35 | powala oznaczyć | pozwala oznaczyć |
| 86 | 32 | Time t2 = new Time(2005, 11 ,18, 45); | Time t2 = new Time(2005, 11, 18, 11, 45); |

| | | | |
|---|---|---|---|
| 126 | 9-10 w części o tytule "Zagnieżdżanie klas" | Klasy zagnieżdżone mają dostęp do wszystkich składowych klasy zawierającej. | Klasy zagnieżdżone mają dostęp do wszystkich statycznych składowych klasy zawierającej. |
| 133 | 3 | Console.WriteLine( "Niejawna konwersja do liczby całkowitej" ); | Console.WriteLine( "Jawna konwersja do liczby całkowitej" ); |
| 135 | 21 | Następnie operator dodaje oba wyniki (6+4), co daje wspólny mianownik sumy (10), oraz mnoży liczniki (2*4), co daje licznik rozwiązania (8). W wyniku otrzymywany jest ułamek 10/8, co jest poprawnym rozwiązaniem. | Następnie operator dodaje oba wyniki (6+4), co daje licznik sumy (10), oraz mnoży mianowniki (2*4), co daje wspólny mianownik rozwiązania (8). W wyniku otrzymywany jest ułamek 10/8, co wyliczeniowo jest poprawnym rozwiązaniem. |
| 141 | 8 | Pozycja Loc1: 200,300 | Pozycja Loc1: 50, 100 |
| 163 | 10 (oraz str. 164 wiersz 26) | Metoda Write interfejsu IStorable w klasie Note! | Metoda Write w klasie Note! |
| 163 | 39 | Nastepnie program tworzy referencję typu interfejsu, na którą za pomocą operatora is rzutowany jest obiekt typu Document: | Nastepnie program tworzy referencję typu interfejsu, na którą za pomocą operatora as rzutowany jest obiekt typu Document: |
| 189 | 11 | Zawartość tablicy Programing_CSharp.Employee[] | Zawartość tablicy ConvertingArrays.Employee[] |
| 198 | 46 | lbt["Hel"] = "do zobaczenia"; | lbt["Wit"] = "do zobaczenia"; |
| 202 | 4, 5 | Wartość: Who Wartość: Is | Wartość: Kim Wartość: Jest |

| | | | |
|---|---|---|---|
| 229 | 40 | Console.WriteLine( "s4: 0, s2: 1, result: 2\n", | Console.WriteLine( "s1: 0, s2: 1, result: 2\n", |
| 282 | 26 | Na końcu program usuwa wybranego delegata za pomocą operatora -+ | Na końcu program usuwa wybranego delegata za pomocą operatora -= |
| 284 | 29-34 | Łańcuch znaków przekazany do delegata Writer<br>Łańcuch znaków przekazany do delegata Logger<br>Łańcuch znaków przekazany do delegata Transmitter | Łańcuch znaków przekazany do metody Writer<br>Łańcuch znaków przekazany do metody Logger<br>Łańcuch znaków przekazany do metody Transmitter |
| 327 | 28 | FillDirectoryTree(tvwTarget, false); | FillDirectoryTree(tvwTargetDir, false); |
| 362 | 42 | DataTime.Now.ToString() | DateTime.Now.ToString() |