



Podstawowe komponenty platformy .NET.

Jakub Zapła 5D



Co to jest platforma .NET.

NET to platforma open-source do tworzenia aplikacji desktopowych, internetowych i mobilnych, które mogą działać natywnie w dowolnym systemie operacyjnym. System .NET obejmuje narzędzia, biblioteki i języki, które obsługują nowoczesne, skalowalne i wydajne tworzenie oprogramowania. Aktywna społeczność programistów utrzymuje i wspiera.



.NET. Komponenty

Przykładowo zainstalowane standardowo komponenty:

- Microsoft Visual C++ 2010 x64 Redistributable - x64 10.0.40219.,
- Microsoft SQL Server Compact 4.0 Web Tools ENU ,
- Microsoft ASP.NET MVC 1.0 ,
- Microsoft .NET Framework 4.5 - 4.8 ,
- IIS URL Rewrite Module 2 ,
- MySQL Connector Net ,
- Microsoft Access database engine 2010 (English) .



Ekosystem platformy .NET

Istnieje wiele wariantów platformy .NET, z których każda obsługuje inny typ aplikacji. Przyczyną wielu wariantów jest część historyczna, część techniczna. Implementacje platformy .NET:

- .NET Framework — oryginalna platforma .NET. Zapewnia dostęp do szerokich możliwości systemów Windows i Windows Server. Jest aktywnie obsługiwana w konserwacji.
- Mono — oryginalna społeczność i platforma .NET typu open source. Implementacja międzyplatformowa programu .NET Framework. Aktywnie obsługiwane w systemach Android, iOS i WebAssembly.
- .NET (Core) — nowoczesna platforma .NET. Międzyplatformowa i open source implementacja platformy .NET, która powinna być zgodna z wiekiem chmury, a jednocześnie znacznie zgodna z programem .NET Framework. Aktywnie obsługiwane w systemach Linux, macOS i Windows



.NET. logo





Omówienie biblioteki klas platformy .NET.

Interfejsy API platformy .NET obejmują klasy, interfejsy, delegaty i typy wartości, które przyspieszają i optymalizują proces programowania oraz zapewniają dostęp do funkcji systemu. Aby ułatwić współdziałanie między językami, większość typów platformy .NET jest zgodna ze specyfikacją CLS i dlatego może być używana z dowolnego języka programowania, którego kompilator jest zgodny ze specyfikacją języka wspólnego (CLS). Typy platformy .NET są podstawą, na której są tworzone aplikacje, składniki i kontrolki platformy .NET. Platforma .NET zawiera typy, które wykonują następujące funkcje:

- Reprezentują podstawowe typy danych i wyjątki.
- Hermetyzowanie struktur danych.
- Wykonaj operacje we/wy.
- Uzyskaj dostęp do informacji o załadowanych typach.
- Wywoływanie kontroli zabezpieczeń platformy .NET.



Zastosowanie

Platforma .NET udostępnia bogaty zestaw interfejsów, a także klasy abstrakcyjne i betonowe (nie abstrakcyjne). Możesz użyć klas betonowych, jak to jest lub, w wielu przypadkach, pochodnych własnych klas od nich. Aby użyć funkcji interfejsu, można utworzyć klasę, która implementuje interfejs, lub uzyskać klasę z jednej z klas platformy .NET implementujących interfejs.



Struktury danych

Platforma .NET zawiera zestaw struktur danych, które są koźmi roboczymi wielu aplikacji platformy .NET. Są to głównie kolekcje, ale także inne typy.

- Array — Reprezentuje tablicę silnie typowanych obiektów, do których można uzyskać dostęp za pomocą indeksu. Ma stały rozmiar na jego konstrukcję.
- List<T> — Reprezentuje silnie typowaną listę obiektów, do których można uzyskać dostęp za pomocą indeksu. Jest automatycznie zmieniany zgodnie z potrzebami.
- Dictionary<TKey,TValue> - Reprezentuje kolekcję wartości, które są indeksowane przez klucz. Dostęp do wartości można uzyskać za pośrednictwem klucza. Jest automatycznie zmieniany zgodnie z potrzebami.
- Uri — Zapewnia reprezentację obiektu jednolitego identyfikatora zasobu (URI) i łatwy dostęp do części identyfikatora URI. DateTime - Reprezentuje moment w czasie, zwykle wyrażony jako data i godzina dnia



Interfejsy API narzędzi

Platforma .NET zawiera zestaw interfejsów API narzędzi, które zapewniają funkcje dla wielu ważnych zadań.

- HttpClient — Interfejs API do wysyłania żądań HTTP i odbierania odpowiedzi HTTP z zasobu zidentyfikowanego przez identyfikator URI.
- XDocument — interfejs API do ładowania i wykonywania zapytań dotyczących dokumentów XML za pomocą linQ.
- StreamReader — interfejs API do odczytywania plików.
- StreamWriter — Interfejs API do zapisywania plików.



KONIEC

Pracę wykonał Jakub Zapła 5D