Java - podstawy

Marcin Szupke

Java

Definicja (wikipedia)

Java - współbieżny, oparty na klasach, obiektowy język programowania ogólnego zastosowania. Został stworzony przez grupę roboczą pod kierunkiem Jamesa Goslinga z firmy Sun Microsystems. Java jest językiem tworzenia programów źródłowych kompilowanych do kodu bajtowego, czyli postaci wykonywanej przez maszynę wirtualną. Język cechuje się silnym typowaniem. Jego podstawowe koncepcje zostały przejęte z języka Smalltalk (maszyna wirtualna, zarządzanie pamięcią) oraz języka C++ (duża część składni i słów kluczowych).

Historia

- Java 1.02 (1996) fajna nazwa i logo. Jej używanie dawało dużo radości. Masa błędów. Zdecydowanie najciekawsze są obiekty
- Java 1.1 (1997) więcej możliwości, nieco bardziej przyjazna dla programistów. Staje się bardzo popularna. Lepszy kod do obsługi graficznego interfejsu użytkownika
- Java 2 (wersje 1.2 1.4, 1998 2002) poważny język programowania o olbrzymich możliwościach. Dostępna w trzech wersjach: Micro Edition (J2ME), Standard Edition (J2SE) oraz Enterprise Edition (J2EE). Stała się preferowanym językiem dla aplikacji biznesowych oraz aplikacji dla urządzeń przenośnych

Historia

- Java 5.0 (wersja 1.5 i kolejne, 2004) ma jeszcze większe możliwości i w większym stopniu ułatwia pisanie aplikacji. W Javie 5.0 oprócz dodania ponad tysiąca nowych klas wprowadzono także zmiany w samym języku, które teoretycznie miały ułatwiać życie programistom i rozszerzyć go o nowe możliwości znane z innych, popularnych języków programowania
- Java 5.0 doczekała się 85 aktualizacji, z których ostatnia została wydana w kwietniu 2015 roku

Jak działa Java?



Stworzenie dokumentu źródłowego w języku Java

Przetwarza dokument i sprawdza, czy w programie nie ma błędów i nie dopuści do jego skompilowania aż do momentu, gdy uzyska pewność, że program uruchomi się poprawnie

Kompilator tworzy nowy dokument, którego zawartość stanowi skompilowany kod bajtowy Javy. Java będzie w stanie zinterpretować ten dokument do postaci, którą będzie w stanie wykonać. Skompilowany kod bajtowy jest niezależny od platformy systemowej

Urządzenia (komputery, smartfony, etc.) nie mają fizycznej maszyny Javy, dysponują jednak maszynami wirtualnymi (zaimplementowanymi w formie programów). Te wirtualne maszyny odczytują i wykonują kody bajtowe

Struktura kodu

plik źródłowy

klasa

metoda instrukcja instrukcja

metoda instrukcja instrukcja

metoda instrukcja instrukcja Plik źródłowy zawiera definicję klasy

Klasa reprezentuje pewien element programu (lub jego całość).

Wewnątrz klasy umieszcza się jedną lub kilka metod.

Metody składają się z instrukcji, które określają sposób działania metody

Anatomia klasy

```
public class Main {
    public static void main(String[] args) {
        System.out.println("Hello, world!");
    }
}
```

Instrukcje

Deklaracje, przypisania, wysłania metod...

```
int x;
String name = "Janek";

x = 4 + 11;
System.out.println("x = " + x);

double d = Math.random();
// komentarz
```

Pętle

```
int x = 11;
while (x > 0) {
    System.out.println("Aktualnie x = " + \underline{x});
   \underline{x} = \underline{x} - \mathbf{1};
for (int i = 0; i < 11; i++) {
    System.out.println("Aktualnie i = " + i);
int y = 11;
do {
    System.out.println("Aktualnie y = " + y);
    y = y - 1;
} while (y > 11);
```

Instrukcje warunkowe

```
int x = 1;
if (x > 0) {
    System.out.println("x jest większe niż 0");
}
if (x > 0) {
    System.out.println("x jest większe niż 0");
} else {
    System.out.println("x jest mniejsze lub równe 0");
}
if (x > 0) {
    System.out.println("x jest większe niż 0");
} else if (x < 0) {
    System.out.println("x jest mniejsze niż 0");
} else {
    System.out.println("x jest równe 0");
}
```

Składnia

- Każda instrukcja musi kończyć się średnikiem
- Komentarze jednowierszowe zaczynają się od dwóch znaków ukośnika
- Zmienne deklaruje się podając nazwę oraz typ
- Definicje klas i metod należy umieszczać wewnątrz nawiasów klamrowych

Testy logiczne

- Testy logiczne przeprowadza się do sprawdzenia wartości zmiennej przy wykorzystaniu operatora porównania, takiego jak:
 - < (mniejszy niż)
 - <= (mniejszy, równy)</p>
 - > (większy niż)
 - >= (większy, równy)
 - == (równy)
 - != (różny)