Java – wprowadzenie

Jędrzej Jajor
Marcin Zienkowicz

Historia Javy

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- 1990 – Bill Joy „Further”
- J. Gosling, P. Naughton, M. Sheridan - Sun Microsystems
- 1991 – projekt Green -> OAK -> Java
- Podobieństwo do języka C++
- Cele: przenośność, łatwość tworzenia aplikacji sieciowych, usunięcie wad C++
- Początkowo: embedded systems
- Nazwa w slangu amerykańskim: kawa

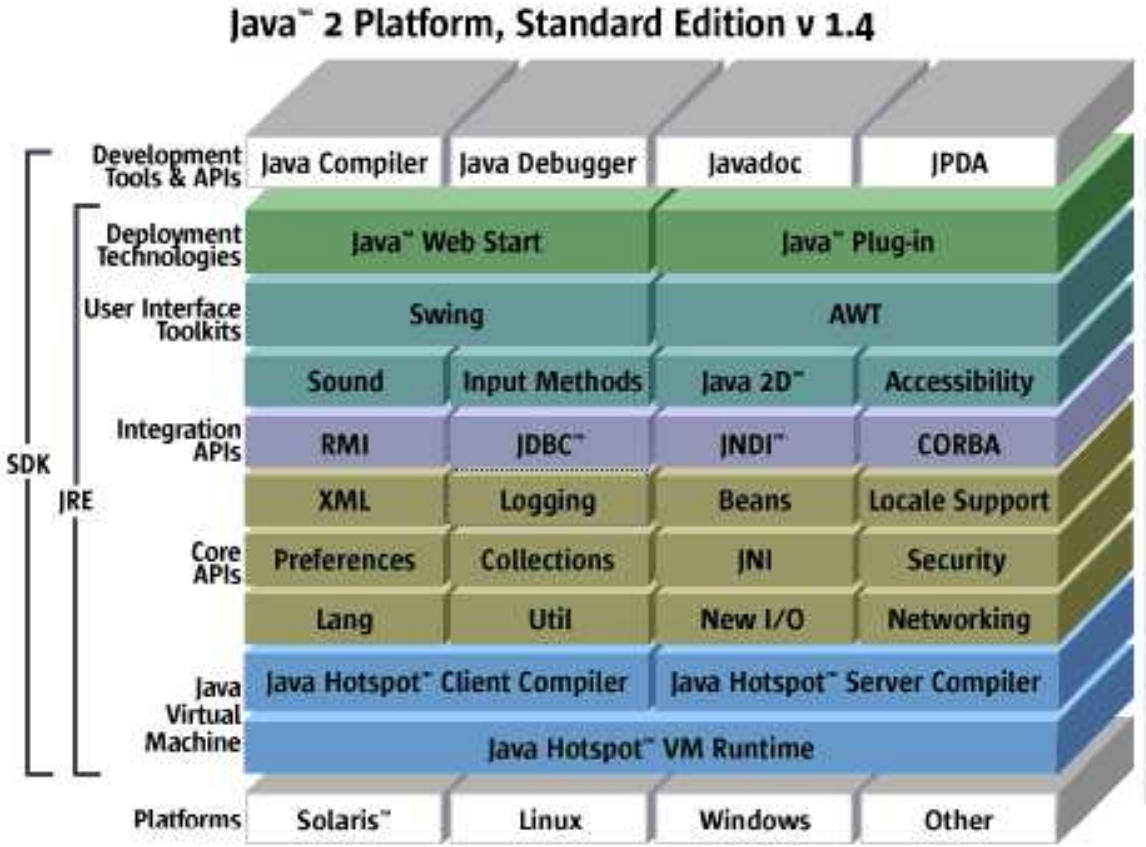
Cechy Javy

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- Obiektowość
- Interfejsy
- Przenośne oprogramowanie
- Byte-code
- Garbage collector
- Aplikacja vs. applet

Java2 SDK

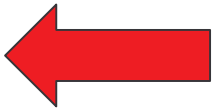
Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie



Świat a model

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

Żaba
płeć gatunek
kumkaj() rechotaj() skacz()



```
class Zaba {
        char plec;
        String gatunek;
        void kumkaj();
        void rechotaj();
        void skacz();
}
```

Klasa a obiekt

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

Monika
żabus leniuchus
kobieta

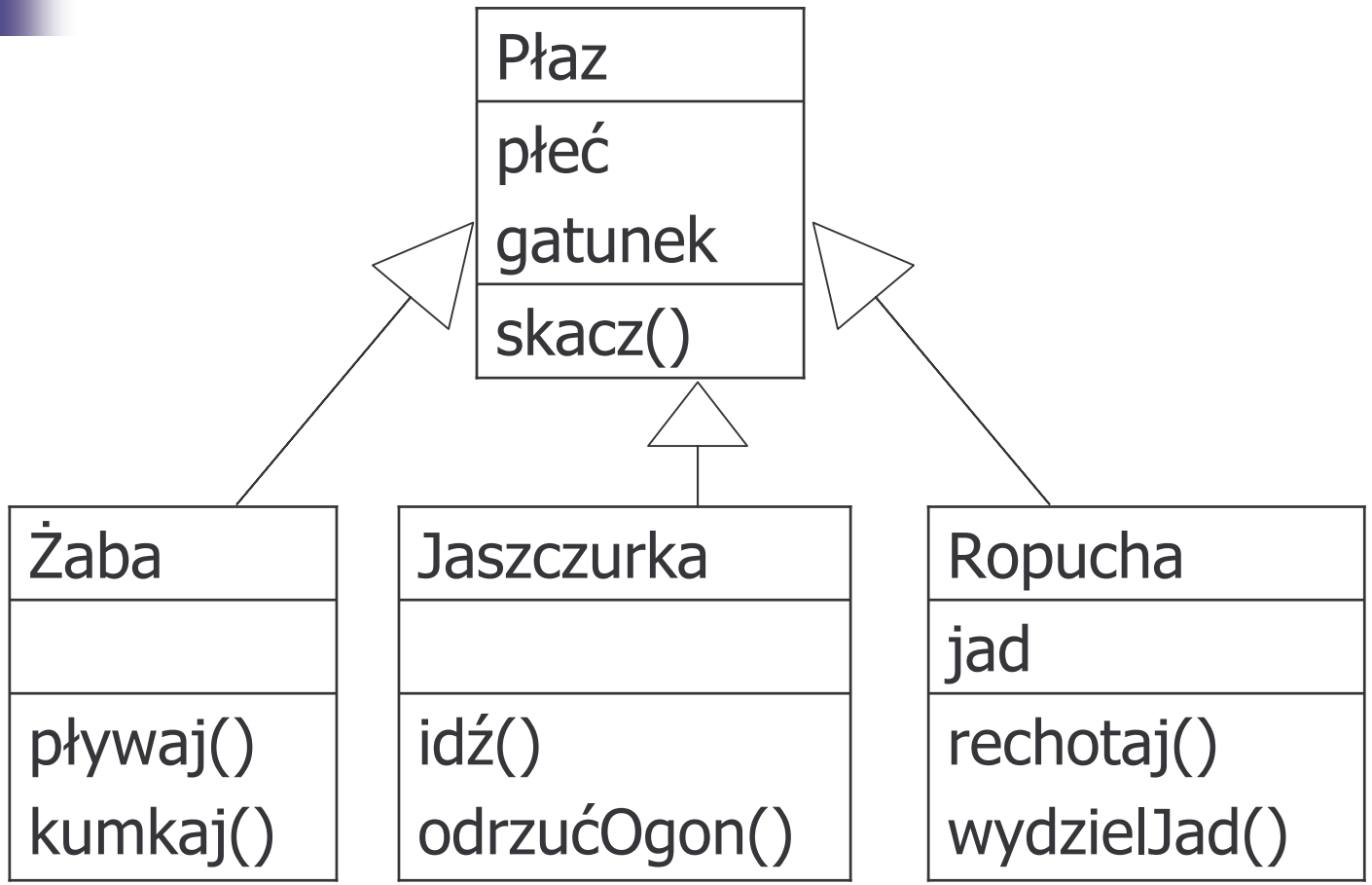
Królewicz
żabus pięknus
mężczyzna

Kermit
żabus pospolitus
mężczyzna

```
Zaba Monika = new Zaba(”żabus leniuchus”, k);
Zaba Krolewicz = new Zaba(”żabus pięknus”, m);
Zaba Kermit = new Zaba(”żabus pospolitus”, m);
```

Dziedziczenie

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie



Dziedziczenie cd.

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

```
class Plaz {
    char plec;
    String gatunek;
    void skacz();
}

class Zaba extends Plaz {
    void plywaj();
    void kumkaj();
}
```

```
class Jaszczurka extends Plaz {
    void idz();
    void odrzucOgon();
}

class Ropucha extends Plaz {
    String jad;
    void rechotaj();
    void wydzielJad();
}
```

Dziedziczenie cd.

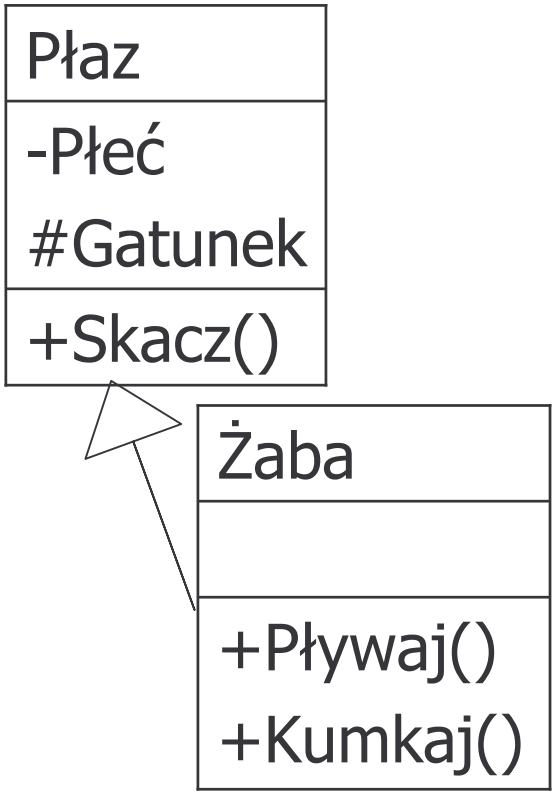
Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- Wszystkie klasy dziedziczą z *java.lang.Object*
- Java wspiera tylko jednokrotne dziedziczenie
- Dziedziczenie wielokrotne może być częściowo zrealizowane dzięki interfejsom
- Metody domyślnie są wirtualne (tzn. wykorzystują polimorfizm)

Hermetyzacja

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

Private(-), protected(#), public(+), default




```
class Plaz {
    private char plec;
    protected String gatunek;
    public void skacz();
}
class Zaba extends Plaz {
    public void plywaj();
    public void kumkaj();
}
```

Abstrakcja

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- Słowo kluczowe abstract oznacza, że dana jednostka (klasa, metoda) fizycznie nie istnieje
  - klasy abstrakcyjne nie mogą mieć obiektów
  - metody abstrakcyjne nie posiadają implementacji
- Klasa z metodą abstrakcyjną musi także być zadeklarowana jako abstrakcyjna

```
abstract public class Figura {}
abstract public void narysuj();
```

Polimorfizm

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- wielopostaciowość
- wiele klas reprezentuje ten sam typ

Polimorfizm cd.

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

```
void Metoda(Plaz x) {
...
}
...
Zaba Monika;
Jaszczurka Konrad;
Plaz Zwierzak;
...
```

```
Metoda(Monika);
Metoda(Konrad);
...
Zwierzak = Monika;
Zwierzak.skacz();
...
Zwierzak = Konrad;
Zwierzak.skacz();
```

Interfejsy

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- Deklaracje metod i definicje stałych
- Dany interfejs mogą implementować różne klasy
- Dana klasa może implementować wiele różnych interfejsów

```
interface Wytresowany {
  final int Wiek = 5;
  void dajGlos();
  void przynies();
}
```

```
class Zaba extends Plaz
implements Wytresowany
{... }
class Pies extends Ssak
implements Wytresowany
{... }
```

Interfejsy cd.

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

```
interface Wytresowany {...}
interface Jadowity {...}
class Ropucha extends Plaz implements
    Wytresowany, Jadowity {...}
...
Ropucha zabus;
Jadowity zwierz1;
Wytresowany zwierz2;

zwierz1 = zabus;
zwierz2 = zabus;
```

Cechy Javy

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- Obiektowość
- Interfejsy
- Przenośne oprogramowanie
- Byte-code
- Garbage collector
- Aplikacja vs. Applet

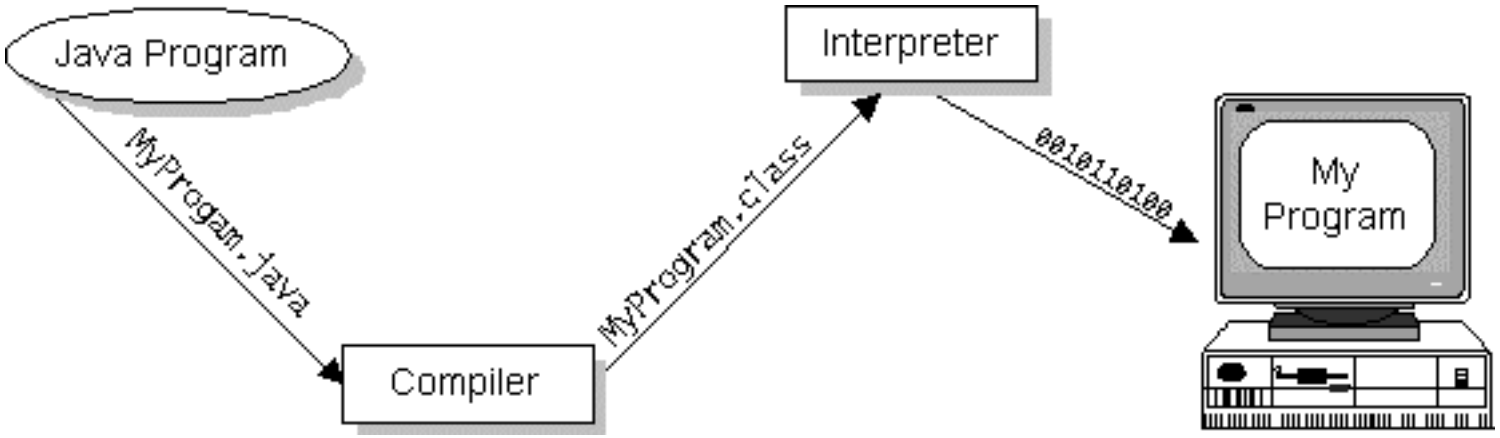
Środowisko

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- JDK (Java Development Kit) i JRE (Java Runtime Environment) – http://java.sun.com
- JVM (Java Virtual Machine), JIT (Just-In-Time)
- Środowiska graficzne, np. Eclipse, JBuilder
- javac - generator bajtkodu dla wirtualnej maszyny Javy
- java – interpreter bajtkodu

Programowanie

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie



Konstruktory

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- służą do tworzenia instancji klas (przydział pamięci i inicjalizacja pól)
- nazwa konstruktora == nazwa klasy
- nie zwracają typu
- mogą być przeciążane
- Jeżeli w klasie nie zdefiniowano wprost żadnego konstruktora, kompilator generuje domyślny konstruktor bezparametrowy

```
public class MojaKlasa {
   public MojaKlasa { // jestem w konstruktorze
   }
}
```

Przykład konstruktora

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

```
public class Okrag {
    public static final double PI = 3.14159;  // stała
    public double r;   // promień okręgu

    public Okrag(double r) {
        this.r = r;
    }
    public Okrag() {
        this(1.0);
    }

    public double obwod() { return 2 * PI * r; }
    public double pole() { return PI * r*r; }
}
```

Zarządzanie pamięcią

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- w Javie nie ma destruktorów
- obiekty, do których nie ma referencji, są usuwane automatycznie przez specjalny wątek JVM: *garbage collector*

Typy prymitywne

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

W Javie wszystko jest obiektem, ale...

- bezpośrednio przechowują pojedynczą wartość
- dane są przechowywane na stosie
- są manipulowane za pomocą wbudowanych operatorów
- nie mogą być rozszerzane przez programistę
- `byte, short, int, long, float, double, char, boolean`
- posiadają odpowiedniki obiektowe (Byte, Short, Integer etc.)

```
int i = 1;
double pi = 3.1415;
boolean isOpen = true;
```

Typy obiektowe

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- przechowują referencję do faktycznego obiektu
- dane są przechowywane na stercie
- są manipulowane za pomocą metod i odwołań do pól
- mogą być swobodnie tworzone i rozszerzane przez programistów

```
MojaKlasa klasa = null;
klasa = new MojaKlasa();
klasa.wykonaj(parametr1, parametr2);
```

Zasięg

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- Tak jak w C/C++, zasięg zmiennej jest określony przez rozmieszczenie nawiasów klamrowych {}.
- Zmienna zdefiniowana w danym zasięgu jest dostępna tylko do końca tego zasięgu

```
{
   int x = 12;   /* tylko x jest dostępne */
   {
        int q = 96;  /* x i q są dostępne */
    }
     /* tylko x dostępne */
     /* q  jest poza zasięgiem */
}
```

```
{
   int x = 12;
   {
        int x = 96; /* błąd */
    }
}
```

Liczby

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- Liczby są reprezentowane przez typy prymitywne o różnej dokładności
  - byte, short, int, long
  - float, double
- Na liczbach prymitywnych działają operatory znane z C/C++ (+-/*=%^)
- Do reprezentacji bardzo dużych liczb służy klasa BigDecimal (zawarta w JDK)

Napisy

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- Napisy są zawsze zapisywane w Unicode
- Klasy reprezentujące napisy
  - java.lang.String – operator '+'
  - java.lang.StringBuffer
- Napisy są obejmowane w podwójne cudzysłowy, pojedyncze znaki – w pojedyncze
- Popularne metody: substr(i, j), indexOf(ch), charAt(n)

```
String napis = „Mój” + „napis”;
String slowo = napis.substr(0, 3);
```

Tablice

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- Tablice są obiektami specjalnego typu
- Indeksy są liczone od 0 i są sprawdzane
- Bardzo ważne pseudopole *length*
- tablice różnych typów są od siebie różne

```
int mojaTablica[ ] = null;
mojaTablica = new int[5];
int mojaTablica2[ ] = {1, 4, 9, 16, 25};
String jezyki [ ] = {"Prolog", "Java"};
```

Wskaźnik a referencja

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- W Javie nie występują wskaźniki takie jak w C/C++
- Wszystkie obiekty są dostępne wyłącznie przez referencje
- Referencje to wskaźnik bez możliwości wykonywania na nim obliczeń
- Adres wskazywany przez referencję odnosi się do sterty i nie może być modyfikowany

```
MojaKlasa klasa = null;
MojaKlasa klasa2 = new MojaKlasa();
```

Stałe

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- Stałą jest zmienna ze słowem final
- Wartość stałej jest ustalana przy pierwszym przypisaniu
- Próba ponownego przypisania wartości jest wykrywana przez kompilator
- Tradycyjnie stałe zapisuje się wielkimi literami

```
final double PI = 3.1415;
final String KLUCZ = „klucz”;
```

Pakiety

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- Rozszerzenie przestrzeni nazw
- Lepsze zarządzanie programem wielomodułowym
- Pakiety ściśle związane z katalogami

```
// plik Zaba.java
package jjmz.zwierz;
public class Zaba {...}

// plik Pies.java
package jjmz.zwierz;
public class Pies {...}
```

```
// plik Zoo.java
import jjmz.zwierz.Zaba;
import jjmz.zwierz.Pies;
// lub
import jjmz.zwierz.*;
public class Zoo {...}
```

Pakiety cd.

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- c:\
  - KlasyJavy
    - jjmz
      - zwierz
        - Zaba.class
        - Pies.class

```
import jjmz.zwierz.Zaba;
import jjmz.zwierz.Pies;
// lub
import jjmz.zwierz.*;
```

- Zmienna CLASSPATH

```
set CLASSPATH=c:\KlasyJavy;c:\jdk\lib;.
```

Pakiety wbudowane

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

Biblioteka klas dostarczana wraz z JDK

- java.util.* - dynamiczne struktury danych
- java.util.zip.* – obsługa plików ZIP
- java.text.* - przetwarzanie tekstów
- java.io.* - wsparcie operacji we/wy
- java.net.* - komponenty sieciowe
- javax.swing.* - komponenty graficzne
- ...

JAR-y

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- Gotowe, spakowane komponenty
- Ułatwienie rozprowadzania dodatkowych pakietów
- Dołączenie do CLASSPATH

  CLASSPATH=c:\KlasyJavy\zwierz.jar
- JAR a ZIP

HelloWorld

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

```
package moj.pakiet;

import inny.pakiet.PewnaKlasa;
import jeszcze.inny.pakiet.*;

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

HelloWorld cd.

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

```
C:\> javac HelloWorld.java
C:\> java HelloWorld
Exception in thread "main"
java.lang.NoClassDefFoundError:
HelloWorld
C:\> echo %CLASSPATH%
C:\JavaSoft\JRE\1.3.1\lib\ext\QT
Java.zip
C:\> set classpath=%classpath%;.
C:\> java HelloWorld
Hello World!!!
```

HelloWorld cd.

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- Pakowanie do JAR

```
jar cvf Hello.jar HelloWorld.class
```

- Dodawanie do CLASSPATH

```
set classpath=c:\KlasyJavy\Hello.jar
```

- Uruchomienie

```
java HelloWorld
```

lub

```
java –jar Hello.jar
```

Java a C++

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- **nie ma typedef** (są klasy),
- **nie ma preprocesora (np. #define** (są stałe)**),**
- **nie ma plików nagłówkowych** (kompilator generuje z plików źródłowych pliki binarne z niezbędnymi informacjami),
- **nie ma struktur i unii** (są klasy),
- **nie ma funkcji** (są metody),
- **nie ma wielodziedziczenia klas** (jest wielodziedziczenie interfejsów),
- **nie ma instrukcji goto** (choć słowo goto jest zastrzeżone),
- **nie ma przeciążania operatorów** (ale jest dociążanie metod),
- **nie ma niejawnych przekształceń (koercji) typów** (można je zapisywać jawnie),
- **nie ma wskaźników** (są klasy, zmienne których wartościami sa obiekty albo null, tablice, obiekty reprezentujące napisy),
- **nie ma delete** (jest automatyczne odśmiecanie).

Przykłady

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

```
package moj.pakiet;

import inny.pakiet.PewnaKlasa;
import jeszcze.inny.pakiet.*;

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

HelloWorldApplet

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

```
import java.applet.Applet;
import java.awt.Graphics;
/**
 * Applet testowy, plik: HelloWorldApplet.java
 * @author JJ&MZ
 */
public class HelloWorldApplet extends Applet {
    /** Wywoływana domyślnie przy odświeżaniu ekranu
     */
    public void paint(Graphics theGraphics) {
        theGraphics.drawString("Hello World !!!", 0, 50);
    }
}
```

HelloWorldApplet cd.

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

```
<!– HelloWorldApplet.html -->
<html>
<head><title>Testowy applet</title></head>
<body>
<h1>Test appletów</h1>
<applet code="HelloWorldApplet.class"
   width="200" height="100">
</applet>
</body>
</html>
```

HelloWorldApplet cd.

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

```
C:\> javac HelloWorldApplet.java

C:\> appletviewer HelloWorldApplet.html
```

Narzędzia

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- javadoc – automatyczna generacja dokumentacji
- appletviewer – przeglądarka appletów
- javap – dekompilacja klas Javy (*.class -> *.java)
- ...

☺ ☺ ☺ ☺

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- niezależność od platformy dzięki JVM
- prawie czysta implementacja paradygmatu obiektowego
- usunięcie pojęcia wskaźnika
- odzyskiwanie „nieużytków”
- weryfikacja kodu w fazie kompilacji i wykonania
- dynamiczne ładowanie klas

☹ ☹ ☹ ☹

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- brak możliwości przeciążenia operatorów
- brak klas parametryzowanych (templates)

Podsumowanie

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- Java to nie tylko applety
- Java Servlet/JSP - tworzenie aplikacji webowych
- Java JDBC - aplikacje bazodanowe
- Java Swing - aplikacje okienkowe
- Java RMI - aplikacje rozproszone
- Java Enterprise Edition - aplikacje biznesowe (odpowiednik .NET)
- ...

Źródła informacji

Plan:
Historia
Cechy
Obiektowość
-Zgodność
-Klasy i obiekty
-Dziedziczenie
-Hermetyzacja
-Polimorfizm
Interfejsy
Środowisko
Pakiety
HelloWorld!
-Aplikacja
-Aplet
Narzędzia
Podsumowanie

- java.sun.com/j2se/1.4
- www.javasoft.com/j2se/1.4/docs
- www.javasoft.com/docs/books/tutorial
- www.BruceEckel.com - Thinking in Java, Thinking in Patterns
- xml.apache.org
- jakarta.apache.org
- www.jug.poznan.pl
- ...

Dziękujemy