

Ćw.1. Tworzenie prostej aplikacji. Kompilacja i uruchamianie

- a) Utworzyć aplikację (klasę) wyświetlającą napis **Witaj Javo!!!**. Kod źródłowy aplikacji należy umieścić w katalogu roboczym w pliku tekstowym o nazwie **Witaj.java** (UWAGA - rozróżnianie są duże i małe litery):

```
public class Witaj { //początek naszej klasy (aplikacji) Witaj
//poniższa metoda main konieczna jest w każdej aplikacji:
    public static void main(String args[])
    {
        System.out.print("Witaj javo!!!\n");
    } //koniec metody main
} //koniec klasy Witaj
```

- b) uruchomić wiersz poleceń, przejść do folderu z plikiem **Witaj.java** i ustawić zmienne systemowe:

```
path=c:\Program Files\Java\jdk1.6.0_03\bin\                (lub odpowiednio inna)
set classpath=.
```

- c) skompilować plik **Witaj.java** poleceniem: **javac Witaj.java**

- d) uruchomić powstały po kompilacji plik **Witaj.class** poleceniem: **java Witaj**

Ćw.2. Tworzenie prostego apletu. Kompilacja i uruchamianie

- a) Utworzyć aplet (klasę) **ApletWitaj** wyświetlający napis **Witaj Javo!!!**. Kod źródłowy apletu należy umieścić w pliku tekstowym **ApletWitaj.java** :

```
import java.applet.Applet; //import klasy Applet
                                //niezbędnej do utworzenia apletu
import java.awt.Graphics; //import klasy Graphics umożliwiającej
                                //wyświetlanie w oknie apletu
//tworzymy własny aplet, który musi dziedziczyć po zaimportowanej
//powyżej klasie Applet:
public class ApletWitaj extends Applet { //początek klasy (apletu)
//poniższa metoda paint jest jedną z predefiniowanych metod
//apletu -umożliwia rysowanie w oknie apletu na obiekcie
//graficznym klasy Graphics
    public void paint(Graphics g) { //początek metody paint
        g.drawString("Witaj Javo!!!", 50, 25); //wyswietlenie napisu
    } //koniec metody paint
} //koniec naszego apletu ApletWitaj
```

- b) Utworzyć plik **Witaj.html** umieszczający aplet javy na stronie WWW:

```
<html>
<head> <title>Aplet Witaj</title> </head>
<body>
<h1>Aplet Witaj:</h1>
<applet code='ApletWitaj.class' width='200' height='50'>
</applet>
</body> </html>
```

- c) skompilować plik **ApletWitaj.java** (jak w ćw.1)

- d) wyświetlić efekt działania apletu - w okienku przeglądarki otworzyć plik **Witaj.html** lub za pomocą polecenia (w wierszu poleceń):

```
appletviewer Witaj.html
```

Ćw.3. Korzystanie z klasy Graphics

a) Utworzyć aplet, który wypisuje tekst: Witaj Javo!!! oraz podkreśla go linią.

```
import java.applet.Applet;           //import klasy Applet
import java.awt.*;                   //import wszystkich klas pakietu java.awt
/** Klasa rysuj.class implementuje klasę Applet, wyświetla tekst podkreślony linią */
public class rysuj extends Applet
{
    // klasa rysuj definiuje metodę paint() klasy Applet
    public void paint(Graphics g) // argument g klasy Graphics to kontekst graficzny,
    // zawiera domyślne ustawienia kolorów i linii
    {
        g.drawString("Witaj Javo!!!", 100, 50); // wyświetla napis na określonej pozycji
        g.drawLine(100,70,170,70);           // rysuje linię łączącą punkty (50,70,200,70)
    }
}
```

b) stosując metody klasy Applet: **setForeground(Color.red)** i **setBackground(Color.lightGray)** – ustalić szare tło apletu i czerwony kolor pierwszoplanowy.

c) stosując metody klasy **Graphics** (rysujemy na obiekcie **g** klasy **Graphics**) narysować odrębnymi kolorami prostokąt, elipsę i łuk tak jak na rys.1. Kolory predefiniowane to: **black, blue, cyan, darkGray, gray, green, lightGray, magenta, orange, pink, red, white, yellow**.

Kolor pierwszoplanowy dla obiektu **g** można ustawiać za pomocą **g.setColor(Color.magenta)**.

Metody rysowania figur na obiekcie **g**:

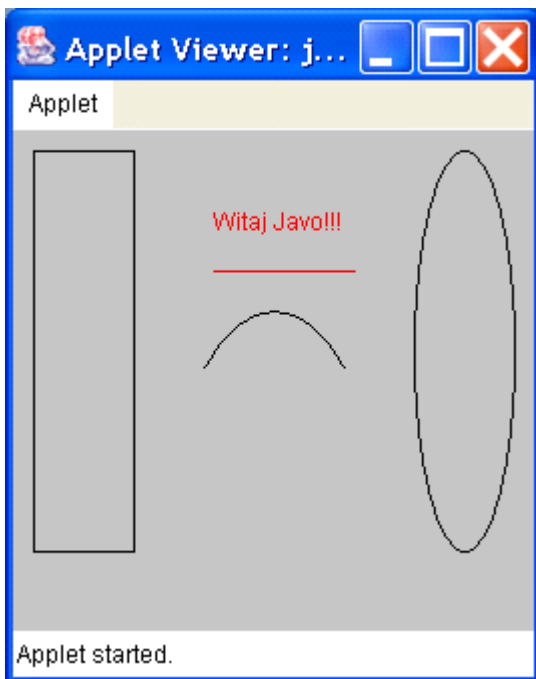
g.drawRect(x, y, szer, wys),

g.drawOval(x, y, szer, wys),

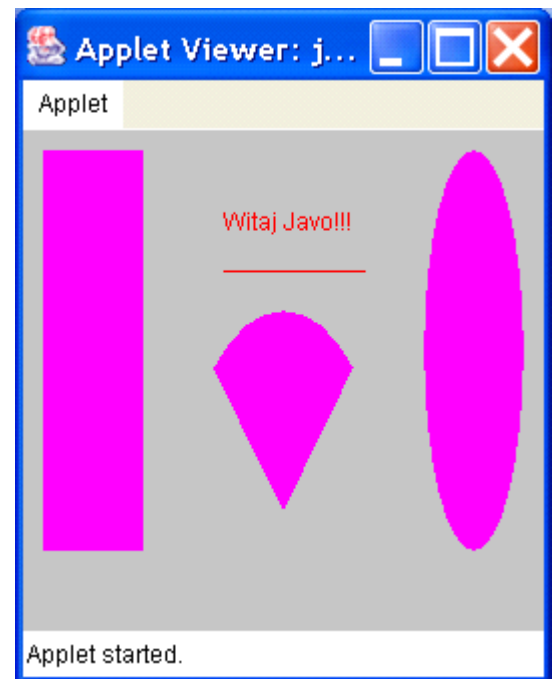
g.drawArc(x,y, szer, wys, ką_t_pocz, szer_kąta)

gdzie **x,y** oznaczają współrzędne lewego górnego wierzchołka okienka, w którym ma być zawarta figura a **wys, szer** oznaczają odpowiednio wysokość i szerokość tego okienka.

d) zmodyfikować aplet tak aby rysował wypełnione kształty odpowiednio za pomocą metod: **g.fillRect**, **g.fillOval**, **g.fillArc** (parametry takie same jak w punkcie c) – rys.2.



Rys.1.



Rys.2.

e) w aplecie narysować dodatkowo kształt zdefiniowany za pomocą tablic współrzędnych:

int [] x={150,190,150,150,170,158,115,110,150};

int [] y={248,248,180,256,267,282,262,240,256};

Zastosować metodę: **drawPolygon(x, y, wymiar_x)** lub
fillPolygon(x, y, wymiar_x)