

Java - wprowadzenie

Tworzenie i uruchamianie projektu w Netbeans

- Uruchom środowisko **Netbeans**. Stwórz nowy projekt typu *Java Application*.

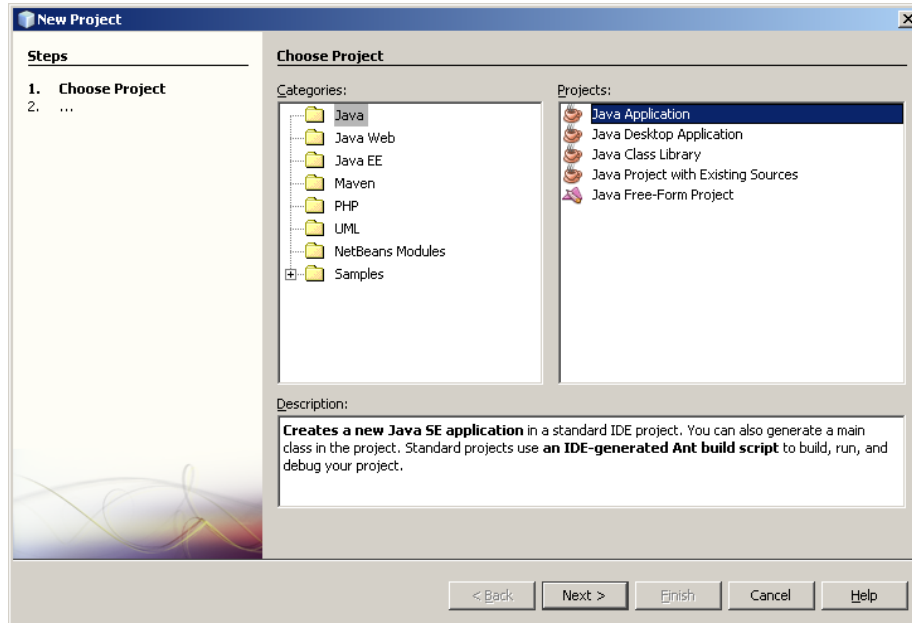


Figure 1:

- Nadaj projektowi nazwę **HelloWorld (Project Name)**, zapisz go na dysku sieciowym (zmień folder w **Project Location**).
- Klasie głównej nadaj nazwę Main (**Create Main Class: helloworld.Main**).
- Przyjrzyj się utworzonej klasie, zwróć uwagę na poszczególne elementy – deklarację pakietu, klasy, metody. Usuń komentarze.
- Uzupełnimy kod, tworząc najprostszy program Hello World. Całość powinna wyglądać następująco:

```
package helloworld;

public class Main {

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

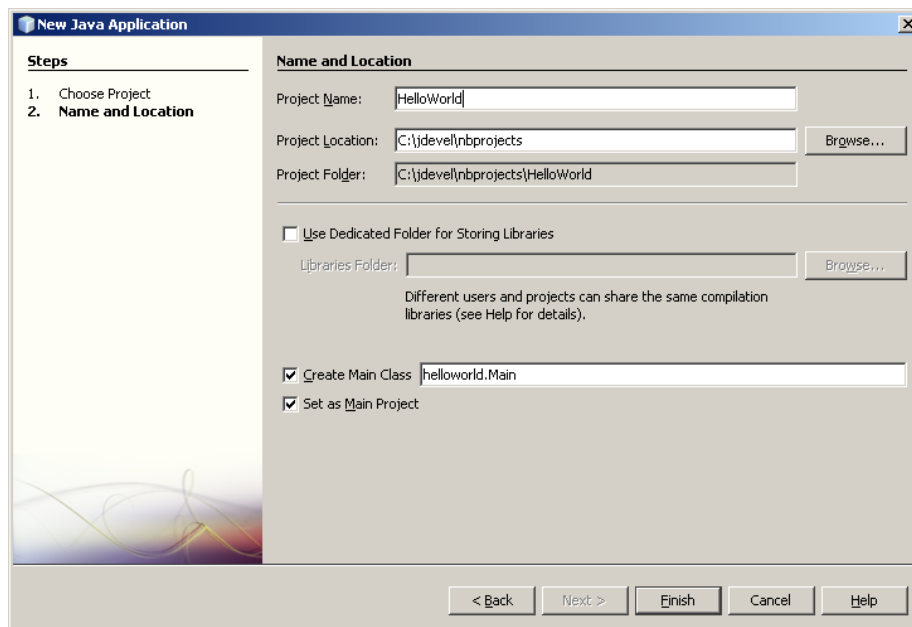


Figure 2:

- Uruchom program naciskając **F6** lub klikając przycisk z zieloną strzałką.
- Wynik działania programu pojawi się w dolnej części ekranu, w okienku **Output**.

Zmienne i wyrażenia

- Zadeklarujemy zmienne rzeczywiste **x** i **y** oraz zmienne całkowite **m**, **n**.
- Zainicjujemy zmienne i obliczymy dwa wyrażenia arytmetyczne z ich udziałem, a następnie wypiszemy wyniki:

```
package helloworld;

public class Main {

    public static void main(String[] args) {
        double x = 3.5, y = 0;
        int m = 3, n = 7;

        double z = x * y;
        int k = m + n;

        System.out.println(z);
    }
}
```

```

        System.out.println(k);
    }
}

```

Instrukcje sterujące

- Za pomocą pętli for wypiszemy liczby całkowite 1..10 oraz ich kwadraty:

```

for(int i=1; i<=10; i++) {
    int kw = i*i;

    System.out.println(i + " - " + kw);
}

```

- Zmodyfikujemy powyższy kod – wypiszemy tylko kwadraty liczb podzielnych przez 3.
- Aby sprawdzić, czy liczba jest podzielna przez 3, wykorzystamy instrukcję warunkową if oraz operator reszty z dzielenia %:

```

for(int i=1; i<=10; i++) {
    int r = i % 3;

    if(r==0) {
        int kw = i*i;
        System.out.println(i + " - " + kw);
    }
}

```

Operacje na tablicach

- Zadeklarujemy dwie nowe zmienne – tablicę liczb całkowitych i tablicę liczb rzeczywistych. Zainicjujemy je, tworząc puste tablice o rozmiarze 10 elementów.

```

double liczbyR[] = new double[10];
int liczbyC[] = new int[10];

```

- W pętli wypełnimy tablice liczbami losowymi. Wykorzystamy funkcję Math.random() oraz rzutowanie typów:

```

for(int i=0; i<10; i++) {
    liczbyR[i] = Math.random();
    liczbyC[i] = (int)(Math.floor(Math.random() * 100));
}

```

- Zawartość tablic wypiszemy na ekranie:

```

for(int i=0; i<10; i++) {
    System.out.println(liczbyC[i]);
}

```

```

    }

    for(int i=0; i<10; i++) {
        System.out.println(liczbyR[i]);
    }

```

- Możemy też użyć pętli for w stylu forEach:

```

for(int liczba : liczbyC) {
    System.out.println(liczba);
}

for(double liczba : liczbyR) {
    System.out.println(liczba);
}

```

- W klasie `Arrays` znajdują się funkcje pomocnicze do operacji na tablicach, między innymi `sort()` oraz `toString()`.
- Posortujemy zawartość obu tablic wykorzystując metodę `sort`:

```

Arrays.sort(liczbyR);
Arrays.sort(liczbyC);

```

- Wypiszemy zawartość tablic na ekranie, wykorzystując metodę `toString`:

```

// zamień na łańcuch tekstowy
String sC = Arrays.toString(liczbyC);
String sR = Arrays.toString(liczbyR);

// wypisz na ekranie
System.out.println(sC);
System.out.println(sR);

```

Zadanie 1

Napisz program losujący 6 liczb całkowitych z przedziału $<1; 49>$. Wylosowane liczby zapisz w tablicy, a następnie wypisz na ekranie w kolejności rosnącej.

Obiektowe wrappery typów prostych

Każdemu typowi prostemu w Javie (np. `int`, `double`, `char`) odpowiada pewien typ obiektowy (np. `Integer`, `Double`, `Character`). Przećwiczmy ich użycie.

- Zadeklaruj kilka zmiennych typu prostego:

```

int m = 3;
double x = 2.3;
char c = 'a';

```

- Zadeklaruj kilka zmiennych typu obiektowego:

```
Integer duzyInt = new Integer(17);
Double duzyDouble = new Double(3.14159);
Character duzyChar = new Character('b');
```

- Wypróbuj automatyczną konwersję pomiędzy typami - spróbuj przypisać do zmiennych obiektowych wartości proste i odwrotnie:

```
duzyInt = 3;
duzyDouble = 666.666;
duzyChar = 'c';
```

```
m = duzyInt;
x = duzyDouble;
c = duzyChar;
```

Zadanie 2

Zmień program losujący 6 liczb - wykorzystaj typ obiektowy `Integer` zamiast typu prostego `int`.

Typy wyliczeniowe

Zdefiniujemy typ wyliczeniowy reprezentujący gatunki piwa.

- Kod definiujący typ wyliczeniowy `GatunekPiwa` umieść ponad klasą `Main`, poniżej listy importów.

```
enum GatunekPiwa {
    LAGER, PILZNER, PORTER, STOUT, PIWO_Z_BIEDRONKI
}
```

- Zadeklarujemy zmienną typu `GatunekPiwa` i przypiszemy do niej wartość:

```
GatunekPiwa piwkoNaDzisWieczor = GatunekPiwa.PIWO_Z_BIEDRONKI;
System.out.println(piwkoNaDzisWieczor);
```

- Jeśli chcesz pobrać tablicę wszystkich możliwych wartości enuma, możesz użyć metody `values()`:

```
GatunekPiwa[] gatunki = GatunekPiwa.values();
```

Zadanie 3

Przygotuj generator menu na imprezę.

- Zadeklaruj tablicę elementów typu `GatunekPiwa`, 3-elementową.
- Wylosuj 3 gatunki piwa i zapisz w tablicy
- Przejdź w pętli po tablicy piw. Użyj instrukcji `switch` do wypisania informacji o piwie:
 - dla piwa `LAGER`, `PILZNER` wypisz ‘piwo jasne’
 - dla piwa `PORTER` i `STOUT` wypisz ‘piwo ciemne’

– dla PIWO_Z_BIEDRONKI wypisz ‘piwo tanie’