

# Kaj je `String[] args`

---

Parametri metode `main`

DIRI 2003 – Programski jeziki

# Kvader

---

- Sestavimo program, ki sprejme iz ukazne vrstice tri pozitivna števila in izpiše volumen kvadra s temi stranicami.
- ```
java Kvader 3 4 5  
60
```

# Kvader

```
public class Kvader
{ // izracunamo volumen kvadra
  public static void main(String[] par)
  {
    int a, b, c;
    int volumen;

    a = Integer.parseInt(par[0]);
    b = Integer.parseInt(par[1]);
    c = Integer.parseInt(par[2]);

    volumen = a * b * c;

    System.out.println(" " + volumen);
  }
}
```

# Branje iz ukazne vrstice

- V args dobimo parametre s katerim poženemo program
- Ločeni s presledki
- nizi
- java bla 12 5 ma
  - V args[0] je "12"
  - V args[1] je "5"
  - V args[2] je "ma"
- args.length - length vrne število parametrov (velikost tabele)

## Tabele - length

- Komponenta (lastnost) `length` vrne velikost tabele
  - NI! metoda (kot pri `String`)
- `int[] tabela = new int[20];`
- `tabela.length`
  - Dobimo 20
- `String bla = "bleble";`
- `bla.length()`
  - 6
- Pomni!
  - Pri nizih dolžino zremo z **metodo** `length` / `niz.length()` /
  - Pri tabelah zremo velikost (rezerviran prostor) z lastnostjo `length` / `tabela.length` /

## Izpiši vse argumente

```
public class Program
{ public static void main(String[] param)
  {
    int st_param, i;
    st_param = param.length;
    System.out.println("Program ima " +
      st_param + " parametrov: ");
    i = 0;
    while (i < st_param)
    {
      System.out.println(i + ". param. = " + param[i]);
      i = i + 1;
    }
  }
}
```

[Program.java](#)

## Kontrola napačnega zagona

- Če bo kdo naš program pognal brez argumentov ali z napačnimi argumenti
  - Program se "sesuje"
- Izpis obvestila, kako je potrebno pognati program

```
if (par.length != 3)
{ System.out.println("Klic\n\n java Kvader a b c\n");
  System.out.println("kjer so a b c mere kvadra!\n");
  System.out.println("Podatki so pozitivna cela števila!");
}
```
- Še vedno se bo program "sesul", če bomo dali tri parametre, a napačne vrste
  - Možno lovljenje napak
  - try / catch konstrukcija
    - Oglejte si sami

## Vsebovan

- Sestavimo program, ki iz ukazne vrstice sprejme dva niza in pove, če je prvi vsebovan v drugem
- java Vsebovan prvi drugi  
Niz "prvi" ni vsebovan v nizu "drugi"
- java Vsebovan prav Naprava  
Niz "prav" je vsebovan v nizu "Naprava"
  
- indexOf

# Vsebuje

---

```
public class Vsebuje
{
    public static void main(String[] par)
    {
        if (par[1].indexOf(par[0]) == -1)
        {
            System.out.println("Niz \"" + par[0] +
                "\" ni vsebovan v nizu \"" + par[1] + "\".");
        }
        else
        {
            System.out.println("Niz \"" + par[0] +
                "\" je vsebovan v nizu \"" + par[1] + "\".");
        }
    }
}
```