

# HTML OSNOVE: Zapiski za izpit

---

## 1. Glava dokumenta (Metapodatki)

---

Te značke gredo v `<head>` in niso vidne na strani, so pa nujne za delovanje.

- `<!DOCTYPE html>` : Pove brskalniku, da uporabljaš HTML5. Vedno v prvi vrstici.
  - `<html>` : Glavni okvir za celotno kodo.
  - `<head>` : Vsebnik za vse spodnje metapodatke.
  - `<title>` : Besedilo v zavihku brskalnika. **Zelo pomembno za izpit!**
  - `<link>` : Povezava do zunanjih datotek (npr. CSS).
    - Primer: `<link rel="stylesheet" href="style.css">`
  - `<meta>` : Informacije o strani (nabor znakov, opis).
    - Primer: `<meta charset="UTF-8">` (poskrbi za šumnike).
- 

## 2. Struktura dokumenta (Glavni deli)

---

- `<!-- ... -->` : **Komentar**. Brskalnik ga ignorira. Uporabno za opombe.
  - `<body>` : Tukaj je vsa vsebina, ki jo uporabnik vidi.
  - `<div>` : Splošen vsebnik. Uporabiš ga, ko želiš združiti več elementov v skupino (blok).
  - `<h1>` do `<h6>` : Naslovi. `<h1>` je največji, `<h6>` najmanjši.
- 

## 3. Oblikovanje vsebine (Besedilo in slike)

---

### Povezave in slike

- `<a>` : **Hiperpovezava**.
  - Uporaba: `<a href="url">Besedilo</a>`

- `<img>` : **Slika**. Nima zapiralne značke!
  - Uporaba: `` (alt je obvezen za dostopnost).

## Oblikovanje besedila

- `<p>` : Odstavek.
- `<br>` : Skoči v novo vrstico (prelom).
- `<hr>` : Vodoravna črta.
- `<strong>` : **Krepko** (poudari pomembnost).
- `<em>` : *Ležeče* (poudari naglas).
- `<small>` : Manjši tisk.
- `<span>` : Uporabi se za oblikovanje le dela besedila znotraj stavka.

## Koda in citati

- `<blockquote>` : Za daljše citate (običajno naredi zamik).
- `<code>` : Za krajše delčke programske kode.
- `<kbd>` : Označuje vnos s tipkovnice (npr. `Enter` ).
- `<pre>` : Ohrani vse presledke in nove vrstice (uporabno za ASCII art ali kodo).

---

### Nasvet za izpit

Če navodilo pravi: "pretvori alineje v neurejen seznam", boš uporabila `<ul>` (unordered list) in `<li>` (list item) za vsako alinejo.

# HTML SEZNAMI, TABELE IN POVEZAVE



---

## 1. Seznami (Lists)

## Neurejen seznam ( `<ul>` )

Uporabiš za alineje, kjer vrstni red ni pomemben (npr. pike).

- `<ul>` - začetek seznama.
- `<li>` - posamezna alineja (list item).

## Urejen seznam ( `<ol>` )

Uporabiš, ko potrebuješ številčenje (1., 2., 3...).

- `<ol>` - začetek seznama.
- `<li>` - posamezna točka.

## Opisni seznam ( `<dl>` )

Uporabiš za slovarje ali sezname s pojmi in razlagami.

- `<dl>` - celoten seznam (description list).
  - `<dt>` - izraz/pojem (description term).
  - `<dd>` - opis/definicija pojma (description details).
- 

## 2. Tabele (Tables)

---

Tabele gradimo po vrsticah.

- `<table>` : Glavni vsebnik tabele.
  - `<thead>` : Glava tabele (naslovna vrstica).
  - `<tbody>` : Telo tabele (glavni podatki).
  - `<tfoot>` : Noga tabele (povzetki na dnu).
  - `<tr>` : Vrstica v tabeli (table row).
  - `<th>` : Celica v glavi (odebeljeno, sredinsko - table header).
  - `<td>` : Navadna celica s podatki (table data).
-

### 3. Atributi in Nastavitve

---

#### Splošni atributi

Uporabljamo jih skoraj pri vseh značkah:

- **id** : Unikaten identifikator (samo eden na strani s tem imenom). Uporaben za povezave in CSS.
- **class** : Razred za oblikovanje. Več elementov ima lahko isto klaso.

#### Povezave ( `<a>` )

1. **Zunanja povezava**: `<a href="https://google.com">Besedilo</a>`
2. **Notranja povezava (sidro)**: `<a href="#nek-id">Skoči na del z id-jem</a>`
  - Pogoji: Nek element na strani mora imeti `id="nek-id"`.

#### Tehnične nastavitve (v `<head>` )

- **CSS**: `<link rel="stylesheet" href="style.css">` (poveže tvoj stil s stranjo).
- **Šumniki**: `<meta charset="UTF-8">` (nujno, da se č, š, ž pravilno izpišejo).

### 4. Kako poiskati nove značke?

---

Če na izpitu dobiš neznano značko (npr. `<marquee>` ):

1. **Išči**: "MDN [ime značke]" (npr. *MDN marquee*).
2. **Preveri**: Kaj značka naredi (*marquee* = pomično besedilo).
3. **Uporabi**: Večina značk deluje po sistemu `<značka>vsebina</značka>`.

#### **Pozor pri** `<marquee>`

Ta značka je zastarela (deprecated), a se na izpitih včasih še pojavi za testiranje iznajdljivosti. Danes se za premikanje uporablja CSS animacije.

# CSS IZBIRALCI (Selectors) 🎨

---

CSS pravilo je sestavljeno iz **izbiralca** in **bloka deklaracij**:

```
izbiralec { lastnost: vrednost; }
```

## 1. Osnovni izbiralci

---

### Po znački (Element selector)

- **Kdaj:** Ko želiš oblikovati vse elemente določene vrste (npr. vse odstavke).
- **Primer:** `p { color: blue; }` (vsi `<p>` bodo modri).

### Po razredu (Class selector)

- **Kdaj:** Ko želiš oblikovati skupino elementov, ki imajo enak `class`.
- **Oznaka:** Pika `.` pred imenom.
- **Primer:** `.rdece-besedilo { color: red; }`

### Po identifikatorju (ID selector)

- **Kdaj:** Ko želiš oblikovati natanko en specifičen element na strani.
- **Oznaka:** Lojtrica `#` pred imenom.
- **Primer:** `#glavni-naslov { font-size: 24px; }`

### Univerzalni izbiralec

- **Oznaka:** Zvezdica `*`.
  - **Kdaj:** Izbere čisto vse elemente na strani. Pogosto se uporablja za ponastavitev robov (reset).
  - **Primer:** `* { margin: 0; padding: 0; }`
-

## 2. Napredno kombiniranje

---

### Izbiranje potomcev (Descendant selector)

- **Sintaksa:** `stars otrok` (presledek vmes).
- **Kdaj:** Izbere otroka, ki se nahaja *nekje znotraj* starša.
- **Primer:** `table a { background: #bada55; }`  
(Vse povezave `<a>`, ki so znotraj katerekoli tabele `<table>`.)

### Disjunkcija (Ali / Več hkrati)

- **Sintaksa:** `prvi, drugi` (veja vmes).
- **Kdaj:** Ko želiš isto oblikovanje uporabiti za več različnih stvari.
- **Primer:** `h1, h2, .podnaslov { font-family: sans-serif; }`  
(Velja za *h1 ALI h2 ALI* elemente z razredom *podnaslov*.)

### Konjunkcija (In / Skupaj)

- **Sintaksa:** `značka.razred` (brez presledka!).
- **Kdaj:** Izbere element, ki ustreza **vsem** pogojem hkrati.
- **Primer:** `p.nujno { border: 1px solid red; }`  
(Samo tisti odstavki `<p>`, ki imajo hkrati tudi `class="nujno"`. Če ima `<div>` razred *nujno*, ne bo izbran.)



### Hitri primeri za izpit

---

1. **Navodilo:** "Vsem slikam v div-u z id-jem 'galerija' nastavi rob."  
`#galerija img { border: 1px solid black; }`
2. **Navodilo:** "Vsem naslovom h1 in h2 nastavi zeleno barvo."  
`h1, h2 { color: green; }`
3. **Navodilo:** "Samo povezavam z razredom 'gumb' odstrani podčrtaj."

```
a.gumb { text-decoration: none; }
```

# NAPREDNI CSS IZBIRALCI

---

## 1. Atributni izbiralci

---

Uporabljamo jih, ko želimo ciljati elemente na podlagi njihovih HTML atributov (npr. `src`, `type`, `alt`, `href`).

### Splošni atribut `[atribut]`

- **Kdaj:** Izbere vse elemente, ki sploh imajo ta atribut, ne glede na njegovo vrednost.
- **Primer:** `img[alt] { border: 2px solid green; }`  
(Izbere vse slike, ki imajo nastavljen opis `alt`.)

### Specifična vrednost `[atribut="vrednost"]`

- **Kdaj:** Izbere elemente, ki imajo atribut nastavljen na točno določeno vrednost.
  - **Primer:** `input[type="text"] { background-color: lightgrey; }`  
(Oblikuje samo vnosna polja za besedilo, ne pa npr. gumbov.)
- 

## 2. Psevdo-razredi (Pseudo-classes)

---

Uporabljajo se za oblikovanje elementov v posebnih stanjih ali na podlagi njihovega položaja v strukturi. Prepoznaš jih po **dvopičju** (`:`).

### Dinamična stanja

- `:hover` : Ko uporabnik z miško pride nad element (najpogosteje na povezavah `<a>` ali gumbih).
  - *Primer:* `a:hover { color: orange; }`

## Pozicijski izbiralci (Zelo pomembno za tabele!)

- `:nth-child(even)` : Izbere vse **sode** elemente (2., 4., 6...).
  - *Primer:* `tr:nth-child(even) { background: #f2f2f2; }`
- `:nth-child(odd)` : Izbere vse **lihe** elemente (1., 3., 5...).
  - *Primer:* `tr:nth-child(odd) { background: white; }`
- `:first-child` : Izbere prvi element znotraj starša.
- `:last-child` : Izbere zadnji element znotraj starša.
  - *Primer:* `p:last-child { margin-bottom: 0; }`

---

# CSS LASTNOSTI (Properties)

---

## 1. Dimenzije in Škatlasti model (Box Model)

---

Vsak element v HTML je "škatla". Od znotraj navzven si sledijo:

Vsebina → Padding → Border → Margin.

- **width / height** : Širina in višina elementa (npr. `200px`, `50%`).
- **padding** : Razmik **znotraj** elementa (med vsebino in robom).
- **border** : Rob elementa.
  - *Sintaksa:* `border: <debelina> < slog> <barva>;`
  - *Primer:* `border: 2px solid black;` (če slog ni podan, uporabi **solid**).
- **margin** : Razmik **zunaj** elementa (odmik od sosednjih elementov).

## Smerne nastavitve

Za `padding`, `margin` in `border` lahko nastaviš vsako stran posebej:



`-top`, `-right`, `-bottom`, `-left`.

- *Primer:* `margin-top: 10px;` ali `border-left: 5px solid red;`

---

## 2. Oblikovanje besedila 🖋️

- `color` : Barva besedila.
- `text-align` : Poravnava (npr. `left`, `right`, `center`, `justify`).
- `font-size` : Velikost pisave (npr. `16px`, `1.2em`, `110%`).
- `font-weight` : Debelina pisave (npr. `bold` za krepko ali `normal`).

---

## 3. Ozadja in Barve 🎨

- `background-color` : Barva ozadja elementa.

### Načini zapisovanja barv:

1. **Ime:** `red`, `blue`, `transparent`.
2. **HEX:** `#FF0000` (rdeča), `#00FF00` (zelena).
3. **RGB:** `rgb(255, 0, 0)`.
4. **RGBA (s prosojnostjo):** `rgba(255, 0, 0, 0.5)` (zadnja številka od 0 do 1).

---

## 4. Pozicioniranje in Ostalo 📌

- `float` : Oblivanje elementa. Najpogosteje `left` ali `right`. Uporablja se, da besedilo teče okoli slike.
- `opacity` : Prosojnost celotnega elementa (od `0.0` do `1.0` ali v %).
  - *Primer:* `opacity: 0.9;` (90 % prosojnost).



## Merske enote: Kdaj katero?

- **px (točke)**: Fiksna velikost, najbolj natančna.
- **em**: Relativna enota glede na velikost starša (npr. `2em` je 2x večje od osnove).
- **% (odstotki)**: Relativno glede na razpoložljiv prostor starša.

### ! Hitra rešitev za robove

Če navodilo pravi: "Nastavite rob tabele na 1px rdeče barve", pa ne pove sloga, vedno napiši:

```
border: 1px solid red;
```

## LaTeX: Preambula in Nastavitve

### 1. Osnova in Razred dokumenta

Vsak dokument se začne z `\documentclass`.

- **Sintaksa**: `\documentclass[možnosti]{razred}`
- **Razredi**:
  - `article`: Za običajne dokumente, članke, naloge.
  - `beamer`: Za prezentacije (prosojnice).
- **Možnosti**: Velikost papirja ( `a4paper` ), velikost pisave ( `10pt` , `12pt` ).
- **Primer**: `\documentclass[12pt, a4paper]{article}`

### 2. Nujni paketi za slovenščino

Brez teh paketov šumniki ne bodo delovali pravilno in dokument ne bo v slovenščini.

- `\usepackage[utf8]{inputenc}` : Pravilno kodiranje znakov.
- `\usepackage[T1]{fontenc}` : Pravilno izrisovanje šumnikov.
- `\usepackage[slovene]{babel}` : Nastavi slovenske naslove (npr. "Kazalo" namesto "Contents") in pravila deljenja besed.

---

### 3. Kdaj naložiti kateri paket? (Knjižnice) 📖

---

Naloži samo tiste, ki jih v nalogi potrebuješ:

Paket	Namen
<code>amsmath</code> , <code>amssymb</code>	Napredna matematika in simboli (npr. $\mathbb{R}$ ).
<code>amsthm</code>	Oblikovanje izrekov, definicij in dokazov.
<code>graphicx</code>	Vstavljanje slik ( <code>\includegraphics</code> ).
<code>booktabs</code>	Lepše, profesionalne tabele (ukazi <code>\toprule</code> , <code>\midrule</code> ).
<code>hyperref</code>	Klikljive povezave v dokumentu in kazalu.
<code>tikz</code> , <code>pgfplots</code>	Risalno orodje za grafe in diagrame neposredno v LaTeXu.

---

### 4. Definiranje lastnih ukazov in okolij 🛠️

---

#### Novi ukazi (`\newcommand`)

Uporabno za krajšanje dolgih zapisov.

- **Sintaksa:** `\newcommand{\ime}{vsebina}`
- **Primer:** `\newcommand{\R}{\mathbb{R}}` (zdaj lahko pišeš samo `\R` namesto dolge kode).

## Izreki in definicije ( `\newtheorem` )

Najprej določiš stil, nato pa definiraš okolje.

- **Stili ( `\theoremstyle` ):**

- `plain` : Odebeljen naslov, poševno besedilo (za izreke).
- `definition` : Odebeljen naslov, navadno besedilo (za definicije).
- `remark` : Poševni naslov, navadno besedilo (za opombe).

- **Primer v preambuli:**

```
\theoremstyle{definition}
\newtheorem{definicija}{Definicija}[section]
```

Uporaba v vsebini: `\begin{definicija}` To je definicija...  
`\end{definicija}`



## Struktura tipične datoteke (Copy-Paste)

```
\documentclass[11pt, a4paper]{article}

% Jezikovne nastavitve
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[slovene]{babel}

% Dodatni paketi po potrebi
\usepackage{amsmath, amssymb, amsthm}
\usepackage{graphicx}

% Definiranje okolij
\theoremstyle{plain}
\newtheorem{izrek}{Izrek}

% Definiranje ukazov
```

```
\newcommand{\mnozica}[1]{\{ #1 \}}
```

```
\begin{document}
```

```
% Tukaj se začne vsebina
```

```
\end{document}
```

# LaTeX: Vsebina in Oblikovanje

## 1. Naslov in Struktura

Ti ukazi poskrbijo za uraden izgled dokumenta.

- `\title{Naslov}`, `\author{Avtor}`, `\date{\today}` : Te podatke zapišeš v **preambulo**.
- `\maketitle` : Ta ukaz pa postaviš v **vsebino**, kjer želiš, da se naslov izpiše.
- `\tableofcontents` : Samodejno ustvari kazalo iz vseh naslovov poglavij.
- `\input{datoteka}` : Vključi vsebino druge `.tex` datoteke (npr. poglavje, ki ga pišeš ločeno).

### Razdelki

- `\section{Naslov}` : Glavno poglavje.
- `\subsection{Naslov}` : Podpoglavje.
- `\subsubsection{Naslov}` : Še nižji nivo.

## 2. Oblikovanje besedila

Ukaz	Učinek
<code>\emph{besedilo}</code>	Poudarjeno (običajno poševno).

<code>\textbf{besedilo}</code>	<b>Krepko.</b>
<code>\textit{besedilo}</code>	<i>Poševno.</i>
<code>\texttt{besedilo}</code>	Pisava pisalnega stroja (za kodo).
<code>\url{naslov}</code>	Izpiše klikljiv URL (potrebuješ paket <code>hyperref</code> ).
<code>\noindent</code>	Prepreči zamik prve vrstice v odstavku.

## Poravnava in okolja

- `\centering` : Ukaz za sredinsko poravnavo znotraj okolja (npr. slike).
- `\begin{center} ... \end{center}` : Okolje za sredinsko poravnavo.
- `\begin{verbatim} ... \end{verbatim}` : Izpiše vse znake točno tako, kot so zapisani (tudi LaTeX ukaze).

## 3. Seznami (Lists)

### Neurejen seznam ( `itemize` )

```
\begin{itemize}
  \item Prva točka (alineja).
  \item Druga točka.
\end{itemize}
```

LaTeX: Seznami, Tabele in Slike 

```
### 1. Seznami (Lists)
**Urejen seznam (`enumerate`)**
```latex
\begin{enumerate}
  \item Prva točka (oštevilčeno).
  \item Druga točka.
\end{enumerate}
```

## Tabele ( tabular )

- `\begin{table}[h]` : Vsebnik (floating), ki tabeli doda napis in oznako.
- `\begin{tabular}{|c|r|}` : Določa število stolpcev ( `c` = sredina, `r` = desno, `l` = levo). Navpičnica `|` nariše navpično črto.
- `\hline` : Vodoravna črta.
- `\toprule`, `\midrule`, `\bottomrule` : Profesionalne črte (potrebuješ paket `booktabs`).
- `\multicolumn{2}{c}{Besedilo}` : Združi več stolpcev v enega.

## SLIKE

```
\begin{figure}[h]
  \centering
  \includegraphics[width=0.5\textwidth]{ime_slike}
  \caption{Naslov slike}
  \label{fig:slika1}
\end{figure}
```

## 3. Sklici in Citati (References)

- `\label{oznaka}` : Postaviš k sliki, tabeli ali poglavju za sklicevanje.
- `\ref{oznaka}` : Izpiše številko poglavja/slike.
- `\pageref{oznaka}` : Izpiše številko strani.
- `\eqref{oznaka}` : Za enačbe (doda oklepaje okoli številke).

## Literatura

- `\cite{kljuc}` : Citira vir iz bibliografije.
- `\bibliographystyle{plain}` : Določi stil citiranja.
- `\bibliography{datoteka_z_viri}` : Vključi seznam virov iz `.bib` datoteke.

# LaTeX: Predstavitve (Beamer) in Matematika 🎓

---

# 1. Predstavitve (beamer)

---

## Osnovne nastavitve

- `\usetheme{ImeTeme}` : Nastavi vizualno temo (npr. `Warsaw`, `Madrid`).
- `\begin{frame} ... \end{frame}` : Ustvari novo prosojnico (okvir).
- `\titlepage` : Ustvari naslovno stran (uporabi znotraj `frame`).

## Bloki (Blocks)

- `\begin{block}{Naslov} ... \end{block}` : Navaden moder blok.
- `\begin{exampleblock}{Naslov} ... \end{exampleblock}` : Zeleni blok za primere.
- `\begin{alertblock}{Naslov} ... \end{alertblock}` : Rdeč blok za opozorila.

## Postopno odkrivanje vsebine

- `\pause` : Ustavi prikaz in počaka na klik.
- `\onslide<2->` : Element se pojavi na 2. koraku in ostane.
- `\only<3>` : Element je viden **samo** na 3. koraku.
- `<+>` : Dodaš k seznamu ( `\begin{itemize}[<+>]` ), da se alineje odkrivajo ena po ena.

## Stolpci (columns)

```
\begin{columns}
  \begin{column}{0.5\textwidth}
    Leva stran prosojnice.
  \end{column}
  \begin{column}{0.5\textwidth}
    Desna stran prosojnice.
  \end{column}
\end{columns}
```

## 2. Matematika

---



## Načini zapisa

- **Vrstični (inline):** `$ ... $` ali `$ ... $` (uporabljamo sredi stavka).
- **Prikazni (display):** `$$ ... $$` ali `$$ ... $$` (enačba v svoji vrstici).

## Posebni ukazi in funkcije

- `\frac{števec}{imenovalec}` : Ulomek.
- `\sqrt[koren]{vrednost}` : Koren.
- `\binom{n}{k}` : Binomski simbol.
- **Funkcije:** Vedno z levo poševnico: `\sin`, `\cos`, `\log`, `\lim`, `\max`.
- **Besedilo v matematiki:** `\text{besedilo}` (da pisava ni poševna).

## Pisave v matematiki

- `\mathbb{R}` : Množice (npr. realna števila  $\mathbb{R}$ ).
- `\mathcal{C}` : Pisane črke.
- `\mathrm{abc}` : Pokončna pisava.
- `\mathsf{abc}`, `\mathtt{abc}` : Sans-serif in pisalni stroj.

## Matematična okolja

- `equation` : Oštevilčena enačba.
- `align` : Poravnava več enačb (uporabi `&` za mesto poravnave in `\\` za novo vrstico).
- `multline` : Za dolge enačbe, ki ne grejo v eno vrstico.
- `cases` : Za sisteme enačb ali definicije po delih.
- `*` (npr. `equation*`) : Izklopi avtomatsko oštevilčenje.

## Primer matrike:

```
\begin{pmatrix}
a & b \\
c & d
\end{pmatrix}
```



# Zapiski: Excel in obdelava podatkov

---



## 1. Uvoz CSV datotek

---

CSV (*Comma Separated Values*) so tekstovne datoteke, kjer so podatki ločeni s posebnim znakom.

- **Postopek:** Data -> From Text/CSV.
  - **Ločila (Delimiters):** Najpogostejše sta to vejica ( , ) ali podpičje ( ; ). Excel mora vedeti, kje se konča en stolpec in začne drug.
  - **Decimalke:** Če uvažáš angleške podatke, je decimalno ločilo pika ( . ), v Sloveniji pa veja ( , ). V nastavitvah uvoza (*Advanced*) preveri, da so decimalke pravilno prepoznane, sicer Excel številke ne bo znal seštevati.
- 



## 2. Osnovne formule in funkcije

---

Formule se vedno začnejo z enačajem =.

- **=SUM( range )** : Sešteje vse vrednosti v izbranem območju.
- **=AVERAGE( range )** : Izračuna povprečje.
- **=IF( pogoj ; "DA" ; "NE" )** : Logični test.
  - *Primer:* =IF(A1 > 50; "Opravil"; "Padel")



Osnovna sintaksa

$$f(x) = \text{ImeFunkcije}(\text{argument1}; \text{argument2}; \dots)$$

Ne pozabi: v slovenskem Excelu argumente ločujemo s **podpičjem** ( ; ).

---



## 3. Oblikovanje celic (Formatting)

---

Določa, kako se podatek *vidi*, ne spremeni pa njegove vrednosti.

- **Decimalke:** Desni klik -> `Format Cells` -> `Number` . Nastaviš poljubno število mest za vejico.
  - **Valute:** Izbereš simbol (€, \$, ...).
  - **Odstotki:** Vrednost \$ 0,5 \$ se bo prikazala kot \$ 50 % \$.
- 

## 4. Tabele (Tables)

---

To ni le navadna mreža, ampak objekt ( `Insert` -> `Table` ).

- **Prednosti:** Samodejno oblikovanje vrstic, avtomatski filtri v glavi in “Table Design” zavihek za hitre izračune (npr. *Total Row*).
- 

## 5. Pogojno oblikovanje (Conditional Formatting)

---

Uporabljamo, ko želimo vizualno izpostaviti podatke.

- **Vrednosti nad/pod mejo:** Celice se obarvajo rdeče, če je npr. vrednost  $\$ < 0 \$$ .
  - **Barvne lestvice (Color Scales):** Odlično za toplotne zemljevide (npr. od zelene za visoke do rdeče za nizke vrednosti).
- 

## 6. Vrtilne tabele (Pivot Tables)

---

Najmočnejše orodje za analizo velikih količin podatkov.

- **Kako delujejo:** Podatke “povzamejo”.
- **Glavna polja:**
  1. **Rows (Vrstice):** Kategorije, ki jih želiš analizirati (npr. *Ime izdelka*).
  2. **Columns (Stolpci):** Dodatna delitev (npr. *Mesec*).

3. **Values (Vrednosti):** Številke, ki jih računaš (ponavadi `Sum` ali `Count`).
4. **Filters:** Omejiš prikaz (npr. samo za leto 2023).

### **Nasvet za StackEdit:**

Če želiš, da so enačbe lepše, uporabi LaTeX zapis, npr.  $x = \frac{-b \pm \sqrt{D}}{2a}$ .

# Wolfram Mathematica: Osnove in sintaksa

## 1. Matematične operacije in simboli

Mathematica razlikuje med navadnim vnosom in krajšicami.

- **Potence:** Uporabiš simbol `^` ali krajšico `Ctrl + 6`.
  - Primer:  $x^2$  vpišeš kot `x^2`.
- **Koreni:** Funkcija `Sqrt[x]` ali krajšica `Ctrl + 2`.
  - Primer:  $\sqrt{x}$  vpišeš kot `Sqrt[x]`.
- **Grške črke:** Vpišeš ime črke med dvema "esc" pritiskoma.
  - Primer: `Esc + alpha + Esc` ustvari  $\alpha$ .

### **Bližnjica**

Za ulomke uporabi `Ctrl + /`, kar ti takoj ustvari zapis  $\frac{a}{b}$ .

## 2. Klicanje funkcij

Vse vgrajene funkcije v Mathematici se začnejo z **veliko začetnico**, argumenti pa so v **oglatih oklepajih** `[]`.

- **Vgrajene:** `Sin[x]`, `Log[x]`, `Table[...]`, `Plot[...]`.
  - **Lastne:** Če definiraš svojo funkcijo, jo kličeš enako: `mojaFunkcija[x]`.
- 

### 3. Spremenljivke in shranjevanje

---

Vrednosti shranimo v spremenljivke z operatorjem `=`.

- **Prireditev:** `a = 10` (vrednost 10 se shrani v `a`).
  - **Brisanje:** Če želiš spremenljivko "izprazniti", uporabiš `Clear[a]`.
- 

### 4. Definiranje lastnih funkcij

---

Pri definiranju funkcij uporabljamo operator `:=` (Delayed Assignment) in podčrtaj `_` (Blank) za argumente.

- **Sintaksa:** `imeFunkcije[spremenljivka_] := izraz`
- **Primer z dvema spremenljivkama:**

$$f(x, y) = x^2 + y^2$$

V Mathematici: `f[x_, y_] := x^2 + y^2`

#### Zakaj podčrtaj?

Podčrtaj `x_` pomeni "karkoli vstaviš na to mesto". Brez njega Mathematica išče točno določen simbol `x`.

---

### 5. Iskanje pomoči (Dokumentacija)

---

Če ne veš, kaj funkcija počne ali kakšne argumente sprejme:

- **Hiter info:** Vpiši `?ImeFunkcije` in pritisni `Shift + Enter`. Izpisala se bo kratka definicija.
- **Celotna dokumentacija:** Klikni na majhno puščico ob opisu ali pritisni tipko `F1`, ko si z miško na imenu funkcije.

---

## 6. Pomen barv (Razhroščevanje)

---

Mathematica uporablja barvno kodiranje, da ti pomaga najti napake še preden poženeš kodo:

Barva	Pomen	Razlaga
Modra	Neznani simbol	Simbol še nima določene vrednosti ali definicije.
Črna	Definirana vrednost	Simbol je že zaseden (npr. si mu prej pripisala <code>a = 5</code> ).
Zelena	Vezana spremenljivka	Pojavi se znotraj funkcij ali modulov (npr. <code>x</code> v <code>f[x_]</code> ).

### Pogosta napaka: Črna barva

Če želiš uporabiti `x` kot neznanko v enačbi, pa je `x` **črne barve**, pomeni, da ima od prej shranjeno neko številko. To bo pokvarilo izračun! Uporabi `Clear[x]`.

## Mathematica: Funkcije za analizo in grafiko

---

### 1. Preoblikovanje izrazov

---

Uporabljamo, ko želimo matematični izraz zapisati v drugačni obliki.

- **Expand[izraz]** : Razčleni oklepaje.
    - Primer:  $(x + 1)^2 \rightarrow x^2 + 2x + 1$
  - **Factor[izraz]** : Izpostavi skupni faktor (obratno od Expand).
    - Primer:  $x^2 - 1 \rightarrow (x - 1)(x + 1)$
  - **Simplify[izraz]** : Poskuša najti najkrajšo pot do zapisa. Če ne gre, uporabiš **FullSimplify**.
- 



## 2. Vsote in limite

---

- **Sum[f, {i, min, max}]** : Izračuna vsoto  $\sum$ .
    - Primer: **Sum[i, {i, 1, 10}]** vrne 55.
  - **Limit[f, x -> a]** : Izračuna limito funkcije, ko gre  $x$  proti  $a$ .
    - Primer: **Limit[Sin[x]/x, x -> 0]** vrne 1.
- 



## 3. Odvodi in integrali

---

- **D[f, x]** : Odvod funkcije  $f$  po spremenljivki  $x$ .
  - **Integrate[f, x]** : Nedoločeni integral (brez mej).
  - **Integrate[f, {x, a, b}]** : Določeni integral na intervalu  $[a, b]$ .
- 



## 4. Reševanje enačb in neenačb

---

- **Solve[enačba, x]** : Simbolno reševanje (izid v obliki  $\{x \rightarrow 2\}$ ). Uporablja dvojni enačaj **==**.
- **NSolve[enačba, x]** : Numerično reševanje (izid v decimalnih številkah).
- **Reduce[enačbe && neenačbe, x]** : Najmočnejše orodje. Reši sisteme in upošteva pogoje (npr.  $x > 0$ ).
- **FindRoot[enačba, {x, start}]** : Iskanje ničle v bližini točke **start**, ko **Solve** ne zmore rešiti zapletenih enačb.



## 5. Risanje grafov (Vizualizacija)

Mathematica je tukaj kraljica. Ne pozabi na sintakso: `Plot[funkcija, {spremenljivka, min, max}]`.

Ukaz	Opis	Kdaj uporabiti?
<code>Plot</code>	Navaden 2D graf	Za funkcije oblike $y = f(x)$ .
<code>ContourPlot</code>	Izris nivojnic	Za implicitne funkcije, npr. $x^2 + y^2 = 1$ .
<code>RegionPlot</code>	Risanje območij	Za neenačbe, npr. $x^2 + y^2 < 1$ .
<code>ParametricPlot</code>	Parametrični graf	Ko sta $x$ in $y$ odvisna od parametra $t$ .
<code>Plot3D</code>	3D površine	Za funkcije dveh spremenljivk $z = f(x, y)$ .

! Pomembno pri enačbah

Pri vseh funkcijah za reševanje (`Solve`, `Reduce`, `FindRoot`) moraš za enačbe vedno uporabiti **dvojni enačaj** `==`. En sam enačaj `=` pomeni shranjevanje vrednosti!



Primer za `Plot3D`

```
Plot3D[Sin[x + y], {x, 0, Pi}, {y, 0, Pi}]
```