
MARKDOWN TESTING

Najprej nekaj besedila, da vidimo kako se to vidi... Ok. Tale beseda naj bi bila **BOLD** ali **bold**, nato pa še nekaj v *Italic* in še underline.

SEZNAMI

1. Element 1
2. element 2
3. element 3.

- element 1,
- element 2,
- element 3.

[] element 1,

[] element 2

[x] označen element

V VIM lahko številne elemente, ki jih označimo, avtomatsko povečamo s C-a in zmanjšamo s C-x.

KOMENTARJI IN CITATI

To pa je lahko komentar ali citat neke osebe.

- Tomaž Kušar -

VSTAVLJANJE OPOMB

V besedilo lahko vstavljamo tudi opombe¹. Opombe so koristne v besedilnih dokumentih, ker omogočajo hiter in enostaven dostop do ključnih informacij. Več informacij² pa lahko opišete spodaj.

¹To opombo naredimo tako, da uporabimo [^ime_popmbe].

²Besedilo spodnjih opomb vpišete med besedilo (npr. po odstavku) z ukazom [^footnote] : Razlaga dodatnih informacij.

TABELE

Table 1: Primer tabele. {#tbl:TblExmpl}

Stolpec1	Stolpec2
Vrstica1	Vrstica1
Vrstica2	Vrstica2

ENAČBE

Lahko pišemo enačbe v vrstičnem načinu $to = Vrsticni * nacin$

Ali pa enačbo napišemo v svoji vrstici tako kot na primer za izračun volumna nekega valja z radijem r in višino h :

$$V = \pi * r^2 * h \tag{1}$$

in se seveda lahko na to enačbo tudi sklicujemo z referenco [1].

LINKI

Na datoteke se lahko sklicujemo s povezavami, na primer takole:

dokument

Če se s kurzorjem postavimo na povezavo in pritisnemo

g^f

bo **VIM** sledil temu dokumentu in ga skušal odpreti.

VSTAVLJANJE SLIK

Na sliko pa se lahko sklicujemo takole [`@fig:sklic_na_sliko`]

z latex kodo pa lahko sliko vstavimo takole:

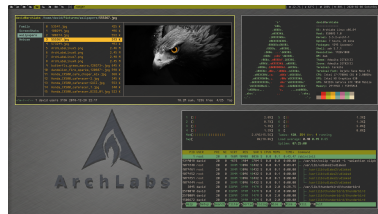


Figure 1: Opis slike.

Z vstavljanjem slike z latex kodo sem imel probleme, ker v templat_u e_i svogel_. latex ni bilo vstavljenega `\package{graphicx}`. Zato sem ga vstavil kar brez kakšnega pogoja.

PROGRAMSKA KODA ali LISTINGS

Takole pa lahko vstavimo programsko kodo, na katero se lahko tudi sklicujemo [{}lst:main]

```
#include <stdio.h>
int main(void)
{
    char greeting[] = "Hello, world";
    printf("%s\n", greeting);
    return 0;
}
```

: Primer programske kode. [{}lst:main]

In programska koda brez specifikacije

To pa je kar tako
nekaj v smislu
pseudo kode ali bloka

V besedilo pa lahko vstavimo tudi kodo iz neke datoteke

```
\lstinputlisting[language=c++, caption={Primer kode.}, label=lst:sklic_na_kodo,
6]]{ime_dateteke_s_kodo.cpp}
```

To je izredno prikladno, saj lahko kodo spremenimo v datoteki in se nato skompila še v besedilu. Kjer se na to kodo lahko tudi sklicujemo [{}lst:sklic_na_kodo].

1: Primer kode.

```
1  #include <stdio.h>
2  int main(void)
3  {
4      char greeting[] = "Hello, markdown.";
5      printf("%s\n", greeting);
6      return 0;
7  }
```

Ta oblika vstavljanja kode nudi nekaj več možnosti prikaza, kot naprimer, da vsatimo le nekatere vrstice iz fila tako, da uporabimo spremenljivko `linenrange={4-6}`

2: Primer iste kode od vrstice 4-6.

```
1      char greeting[] = "Hello, markdown.";
2      printf("%s\n", greeting);
3      return 0;
```

BESEDILO V VEČ STOLPCIH

Pisanje v več stolpcih v markdown ni omogočeno. Zato moramo to narediti z latex kodo. Začnemo z ukazom **`\begin{multicols}{2}`** nato vstavimo nekaj besedila in končamo z ukazom **`\end{multicols}`**. Besedilo se bo avtomatsko razporedilo v dva

stolpca s približno enako besedila v vsakem stolpcu. Omenimo naj še nekaj uporabnih ukazov: **setlength{columnseprule}{0.1pt}** - nariše tanko črto med stolpcema, **null** - zapolni del stolpca s praznim območjem, **vfill**.

Slika v dveh stolpcih

Vstavljanje slike z opisom slike v sosednjem stolpcu je zelo koristno, saj omogoča lažje razumevanje vsebine. Bralci bodo lahko lažje razumeli vsebino, če bodo lahko videli sliko, skupaj z opisom, da bodo razumeli, kaj prikazuje slika. Vstavljanje slike z opisom slike v sosednjem stolpcu je zelo koristno, saj omogoča lažje razumevanje vsebine. Bralci bodo lahko lažje razumeli vsebino, če bodo lahko videli sliko, skupaj z opisom, da bodo razumeli, kaj prikazuje slika.

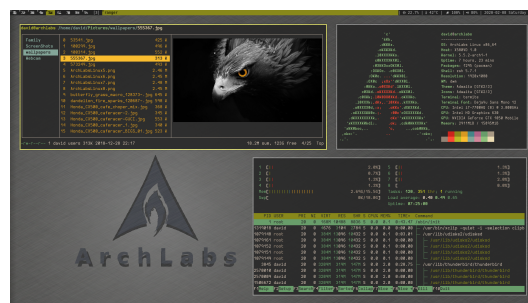


Figure 2: Opis slike.

Programska koda v dveh stolpcih

```
1    int main(){
2        int i = 2;
3        for (i=0;i<5;i++){
4            cout<<"Hello";
5        }
6    }
```

Vstavljanje programske kode z opisom kode v sosednjem stolpcu je koristno, ker omogoča hitrejšje razumevanje kode in jo lažje razvrščanje. Opisi so lahko zelo uporabni, ko poskušamo razumeti kodo, ki smo jo napisali nekaj časa nazaj. Spodbuja tudi dobre kode, saj je lažje opaziti, kaj se dogaja v kodi, ko jo lahko prikažemo skupaj z opisom. Prav tako lahko pomaga poenostaviti kompleksno kodo.