

## Упражнение 06

### Редактор vi.

---

Редакторът Vi (съкращение от visual) е един от най-популярните текстови редактори в Linux света. Можете да го намерите в почти всички Linux дистрибуции, като той работи по един и същ начин на тях.

### Стартиране

---

- **vi** <нов\_файл/съществуващ\_файл>- ако файлът съществува го отваря за редактиране, иначе може да създадете нов файл
- може да стартирате vi и без да му подавате име на файл и да пишете информация в редактора. Тук разликата е, че при записването трябва да укажете име на файла.

### Режим на работа

---

Vi има три режима на работа - команден режим(**command mode**), въвеждащ режим(**insert mode**) и режим промяна(**replace mode**)

### Режим промяна

При стартиране на Vi винаги влизате в този режим. Тук можете да движите курсора, да изпълнявате команди на екрана, чрез които да копирате, променяте, замествате, изтривате текст и други.

### Въвеждащ режим

За да влезете в него е нужно да натиснете клавиша "**a**". След влизането във въвеждащ режим вече можете да пишете и триете.

- **i** - въвеждане на мястото на курсора
- **I** - въвеждане в началото на реда
- **a** - въвеждане след мястото на курсора
- **A** - въвеждане в края на реда
- **o** - добавя нов, празен ред след текущия
- **O** - добавя нов, празен ред преди текущия

**ESC** – преминава се от въвеждащия режим към режим промяна

## Навигация (движение на курсора)

- k - премества курсора нагоре (може да се използва стрелка нагоре)
- j - премества курсора надолу (може да се използва стрелка надолу)
- h - премества курсора наляво (може да се използва стрелка наляво)
- l - премества курсора надясно (може да се използва стрелка надясно)
- ^ - към първият символ от текущия ред
- \$ - към последния символ на текущия ред

## Промяна на текст

- r - заменя символа, на който се намира курсора
- sw - променя думата от текущото положение на курсора до края на думата
- snw – променя n на брой думи от текущата позиция на курсора надясно
- s^ - променя думата от текущото положение на курсора до първия символ на реда
- s\$ - променя думата от текущото положение на курсора до последния символ на реда
- ss - изменя целия ред, на който се намираме
- nss - променя n на брой реда от текущия надолу
- u - премахва последната промяна
- U - премахва абсолютно всички промени

## Изтриване на текст

- x - изтрива символа, на който се намира курсора
- dw - изтрива дума от позицията, на която се намира курсора до края
- dnw - изтрива n думи от текущата дума надясно
- dd - изтрива редът, на който се намира курсора
- ndd – изтрива n на брой реда от текущия надолу
- d^ - изтрива всичко от позицията на курсора до първия символ на реда (включително)

- d\$ - изтрива всичко от позицията на курсора до последния символ на реда (включително)

## Копиране и поставяне в текст

- уw - копира думата
- уу - копира ред
- р - поставя копираните символи

### Примери:

5уw (или у5w) - копира следващите 5 думи

уу и 5р - копира текущия ред и го поставя 5 пъти

## Преместване в текст

- dw + р - преместване на дума
- dd + р преместване на ред

## Търсене в текст

- /низ - търсене на низ надолу от текущия ред във файла
- ?низ - търсене на низ нагоре от текущия ред във файла
- n - търсене на следващ резултат в същата посока
- N - търсене на следващ резултат в предишната посока

## Обединение на текст

- J- долния ред се залепва към горния

## Запис и изход от vi

- :w - записва промените във файла. Ако сме отворили редактора без да указваме име на файл трябва да го укажем тук.
- :q - излиза от редактора
- :q! - излиза от редактора без да запише промените
- :wq - записване и изход

- `:e file` - започва редактирането на нов файл
- `:e! file` - започва редактирането на нов файл, без да се запазват промените по стария редактиран
- `:r file` - прочита файлът `file` и добавя съдържанието му след курсора
- `!:cmd` – изпълнява команда от средата

## Адресация

`: n1,n2` – от ред `n1` до ред `n2`

`:=` - текущия ред

`:. , n2` – от текущия ред до `n2`

`: 1,$` - от първия до последния

`:-2,+5` – 2 реда преди текущия до 5 реда след

**d** – изтриване

**w** – записване

**p** – извеждане

**s** - промяна

Пример:

`:1,$s/int/char/`

`:1,$s/int/char/g`