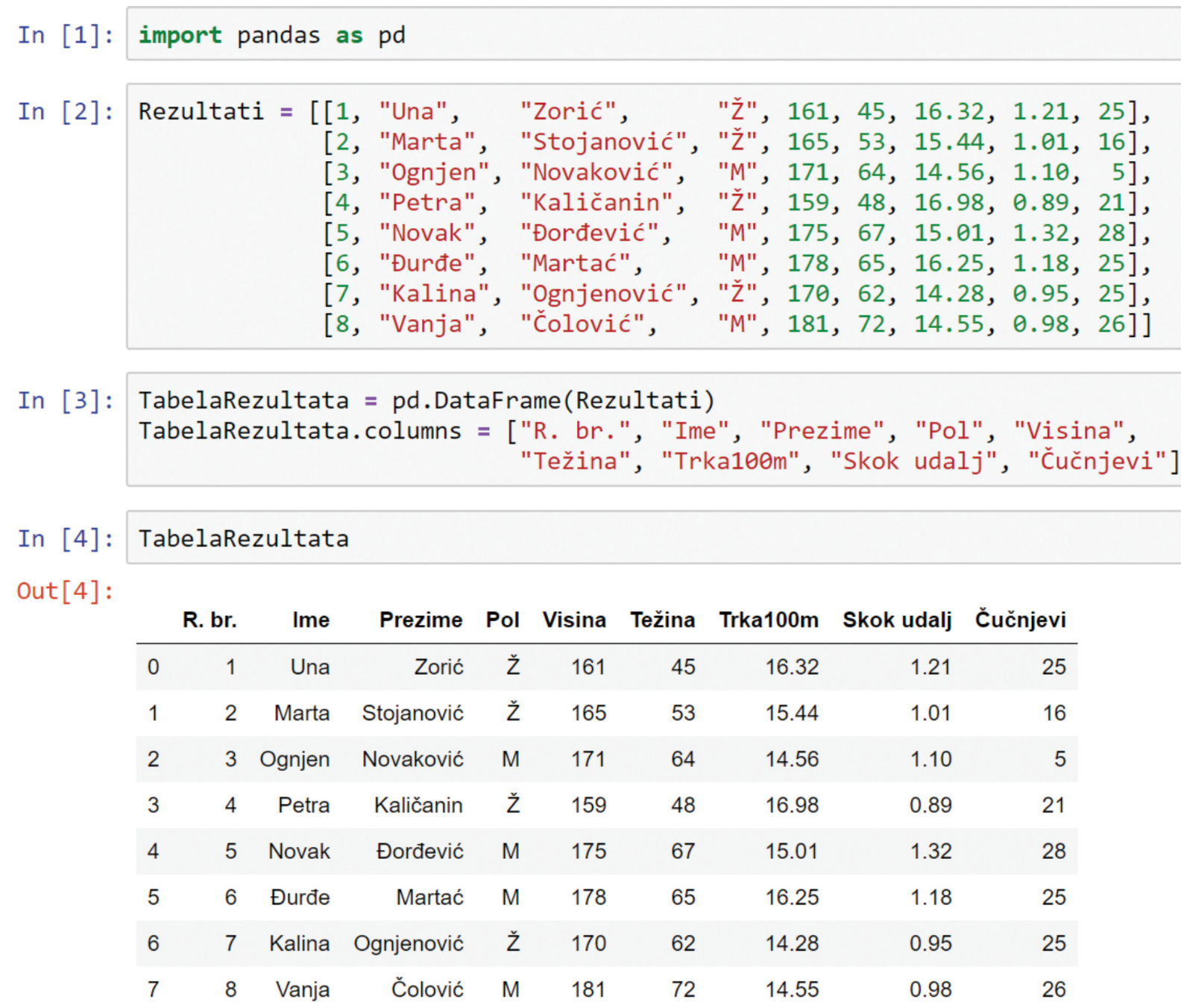
# 28. Сортирање и филтрирање табеле

Све опције које ћемо ти показати у овој лекцији и наредним лекцијама примењиваћемо на проширену **табелу с резултатима такмичења**.

Да би се табела допунила подацима у новим колонама, најпре се мора допунити угњежћена листа ***Rezultati****.*

У сваки елемент листе су на позицијама **индекса 3, 4 и 5 уметнути подаци о полу, висини и тежини такмичара** (слика 3.6.1. ћелија 2).



Потом је од листе направљена **табела** *(Data Frame)* у којој су уместо индекса, **колонама** додељени одговарајући називи (ћелија 3).

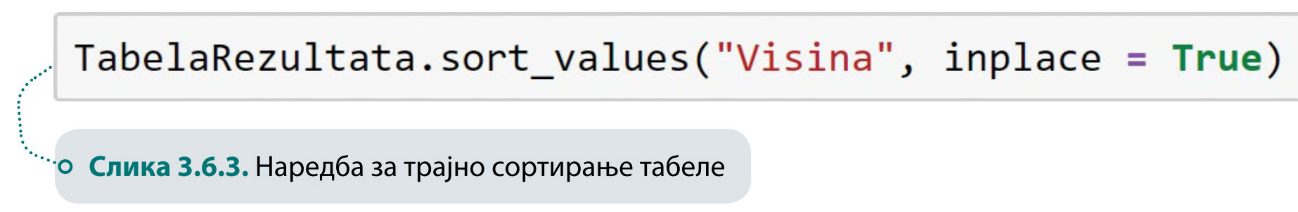
# Сортирање

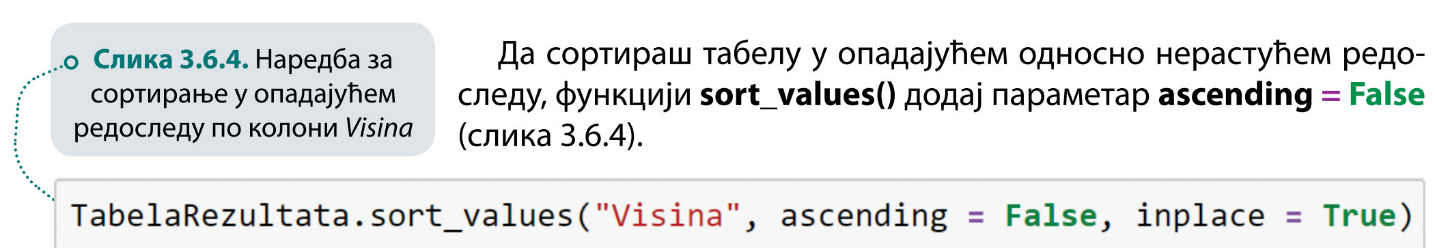
1. За **сортирање** података се користи функција **sort\_values().**

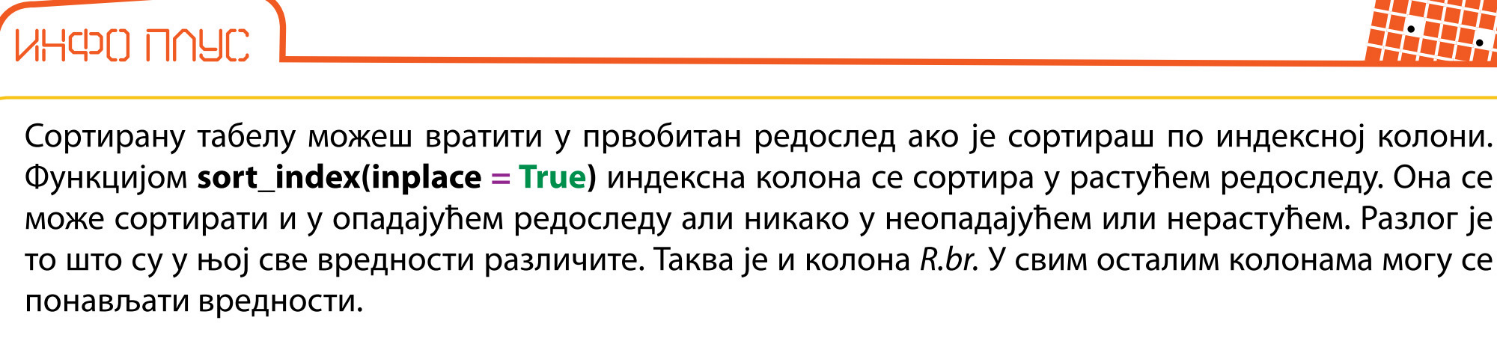
Њен параметар је **колона** по којој хоћеш да сортираш табелу.



На овај начин се добија само **увид у изглед сортиране табеле**.

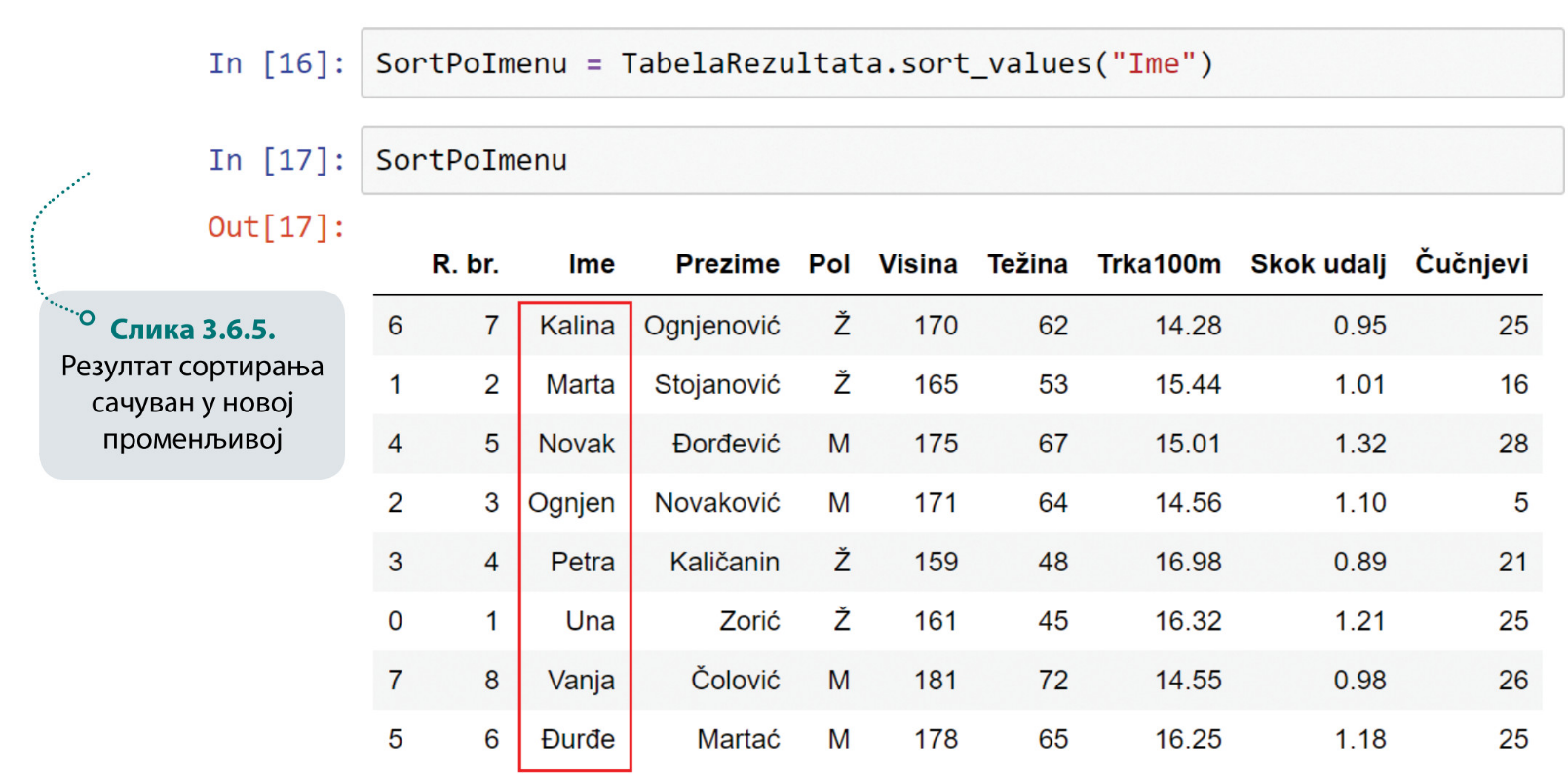
1. Да **трајно сортираш табелу** тј. да резултат сортирања остане запамћен у променљивој у којој је табела сачувана, потребан је параметар **inplace** (на месту, на лицу места). Његову логичку вредносттреба поставити на **True** (слика 3.6.3).` ****
2. **Опадајући редослед**

****



Сортирану табелу је неретко корисно сачувати у **новој променљивој.** На тај начин се касније у програму можеш позвати на њу и поново приказати сортирану табелу. Табелу сачувану у новој променљивој можеш наставити да обраћујеш и анализираш без утицаја на почетну табелу (сачувану у старој променљивој).

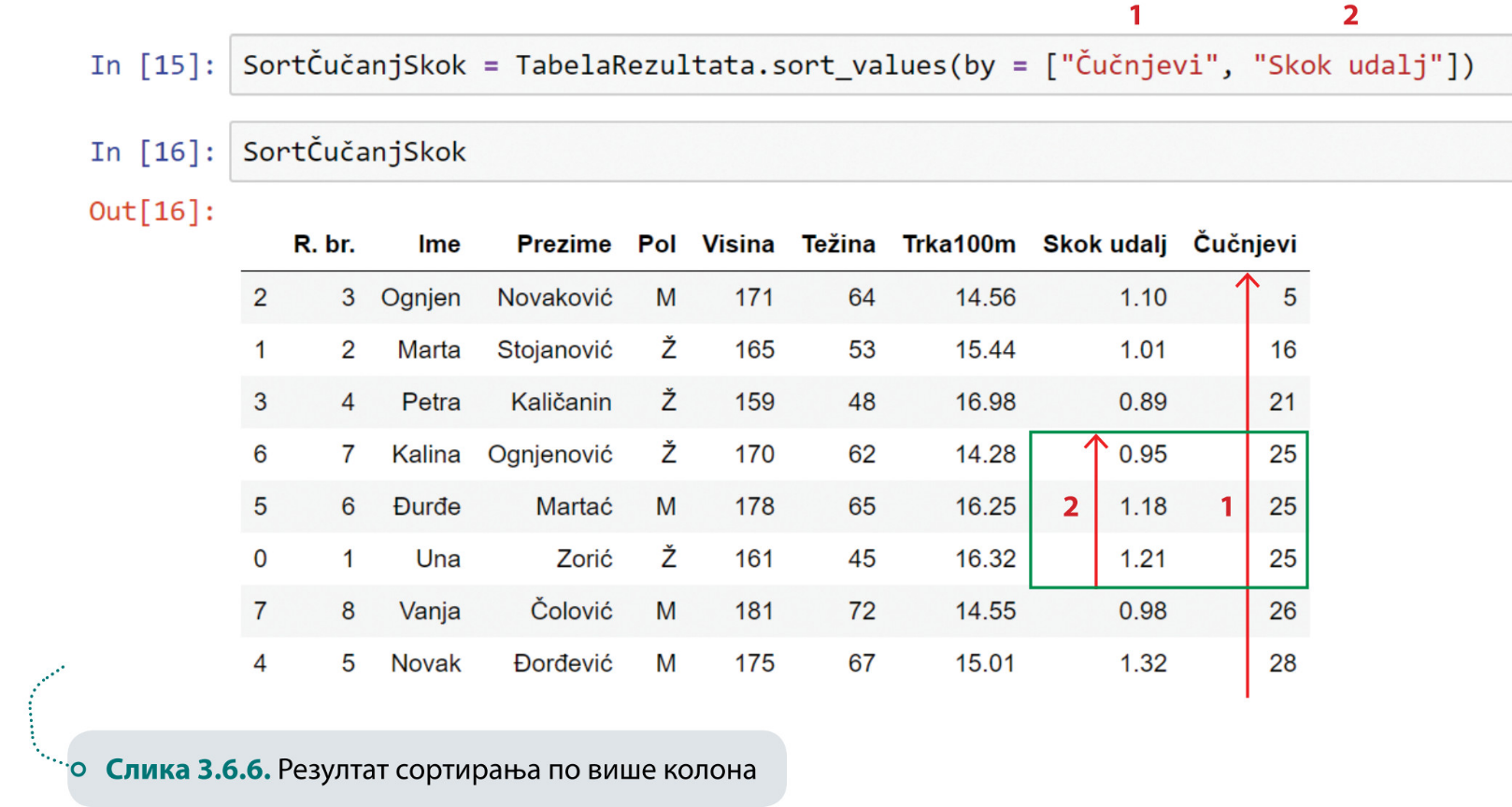
Ово је велика **предност** *Jupyter-a* над на *Ехсе1-ом.* У *Excel-y* би морао/-ла да сачуваш сортирану табелу у новом радном листу или у новој радној свесци.



На слици 3.6.5. је пример у ком је табела с резултатима такмичења *(TabelaRezultata)* сортирана по колони *Ime* у растућем редоследу и таква сачувана у новој променљивој под називом **SortPolmenu.** Када сортирану табелу чуваш у новој променљивој, параметар **inplace** није потребан.

## Сортирање по више колона

Табелу можеш сортирати по две колоне или по више њих.

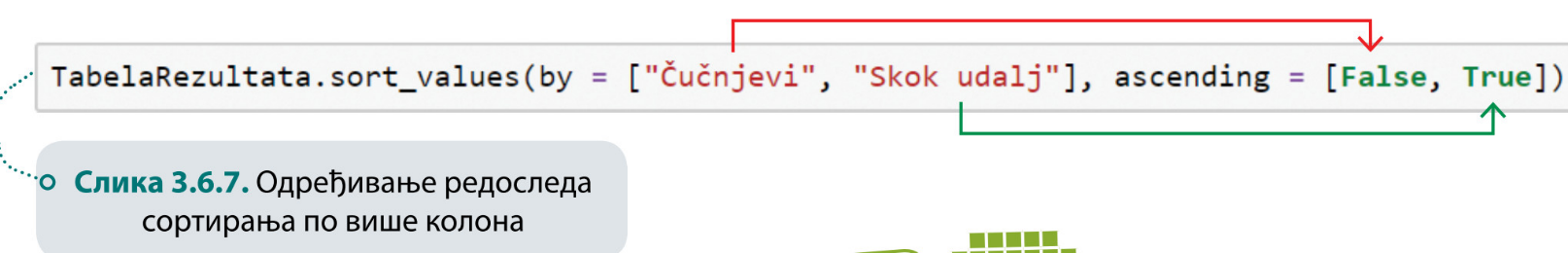


На слици 3.6.6. приказана је наредба за сортирање табеле у неопадајућем редоследу **по колони Čučnjevi, па по колони Skok udalj***.*

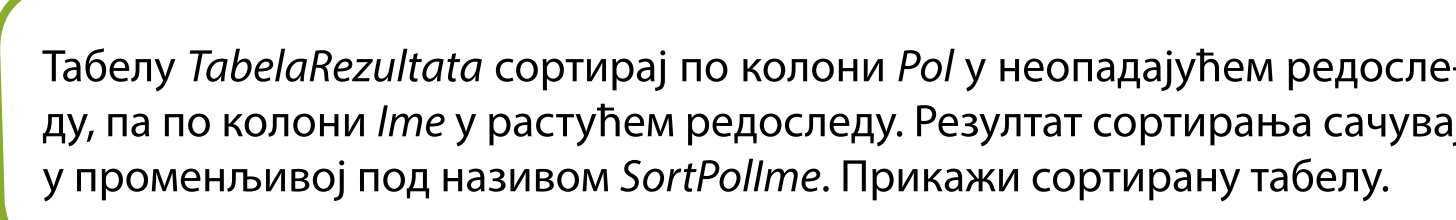
У оваквим случајевима је параметар функције **sort\_values() листа колона**.

Назив листе **(by)** има више значења али се овде преводи као **по** (скраћено од **колоне по којима се табела сортира***).*

**Редослед колона у листи одрећује редослед сортирања табеле**.

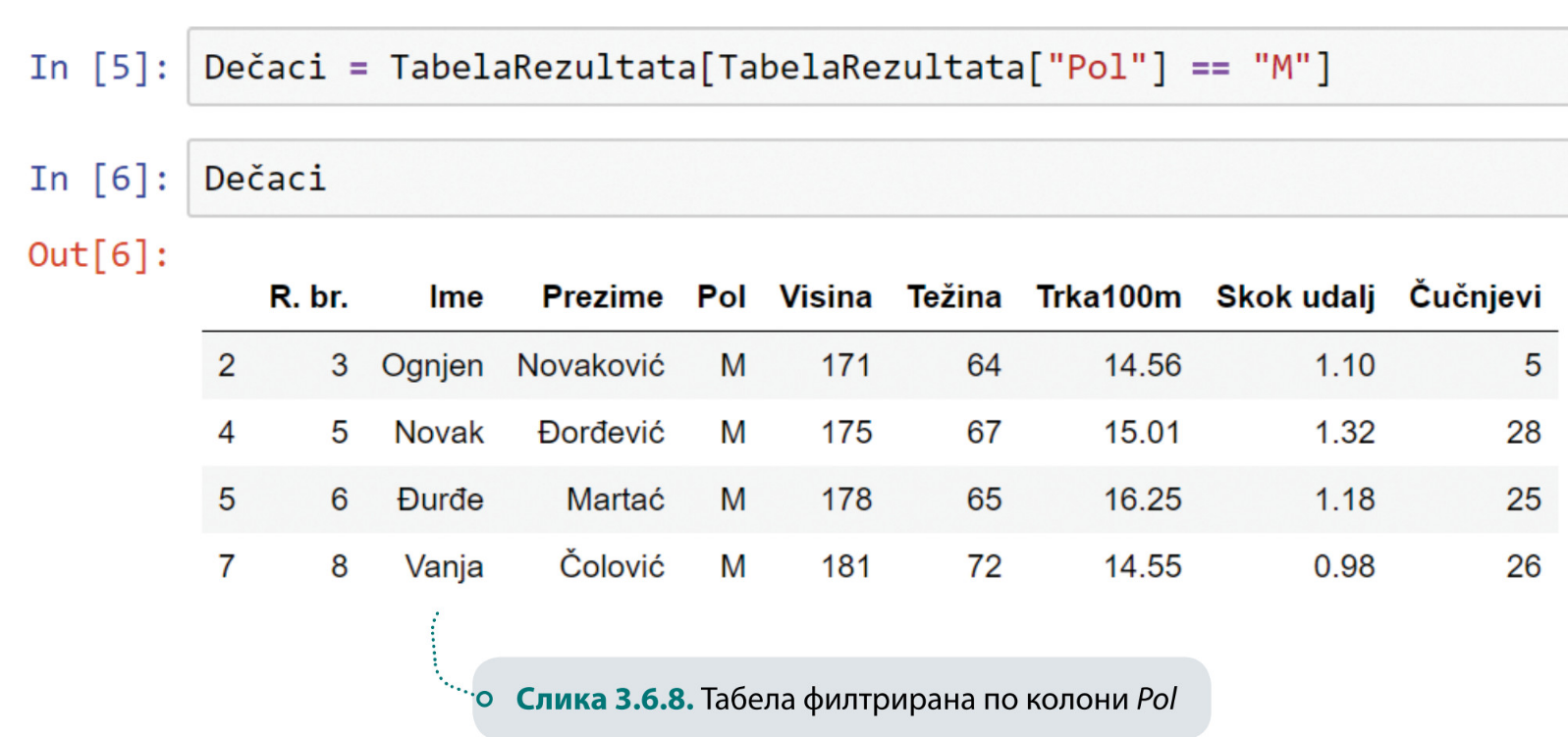


За сваку колону по којој се сортира табела може се одредити **редослед сортирања**. Параметар који то одређује такође се мора навести у **облику листе**. Табела би овако била сортирана по колони *Čučnjevi* у **нерастућем** редоследу, па по колони *Skokudalj* у **неопадајућем** редоследу.



## Филтрирање

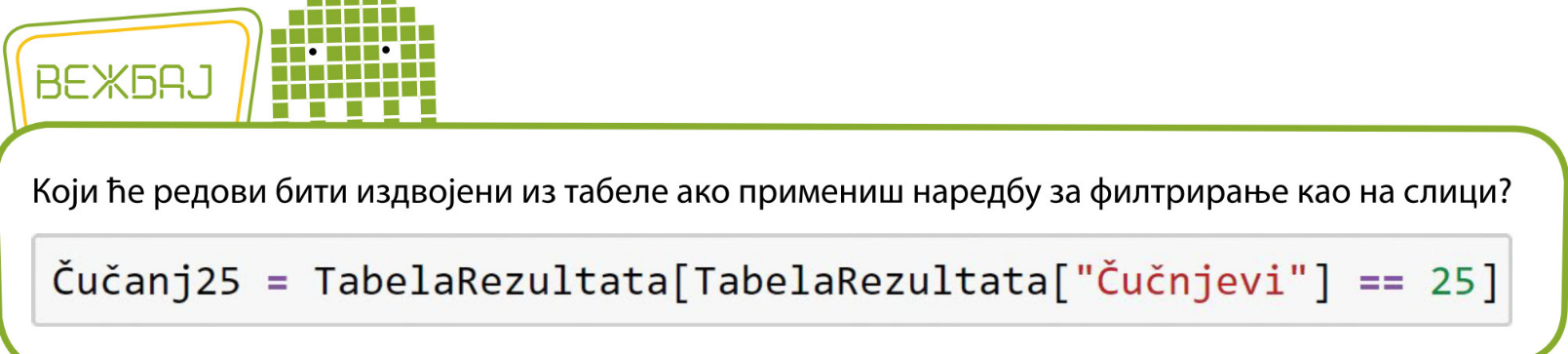
Филтирањем се из табеле **издвајају само они редови који задовољавају задати критеријум**.



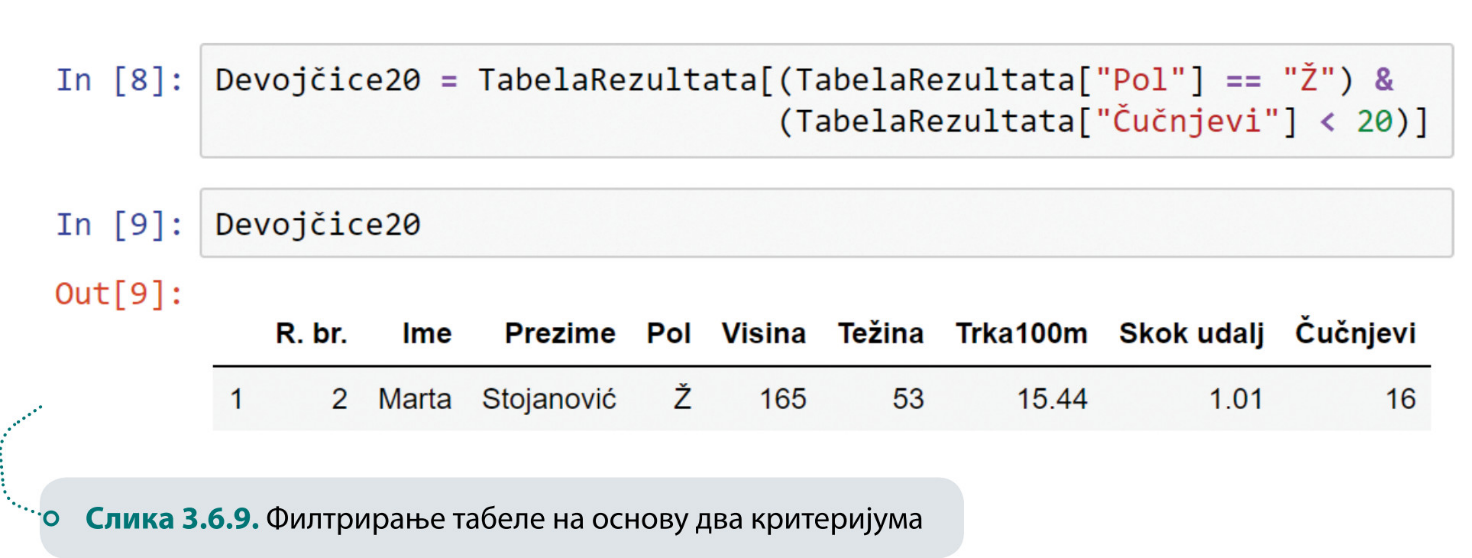
**Критеријуми се уносе у угласту заграду иза назива табеле**.

1. За њихово постављање користе се **оператори поређења**, називи **колона** и **константе**.

Овде се издвајају **само редови у којима су дечаци**, односно у којима су подаци у колони *Pol* једнаки слову М. Резултат филтрирања је сачуван у променљивој *Dečaci.*

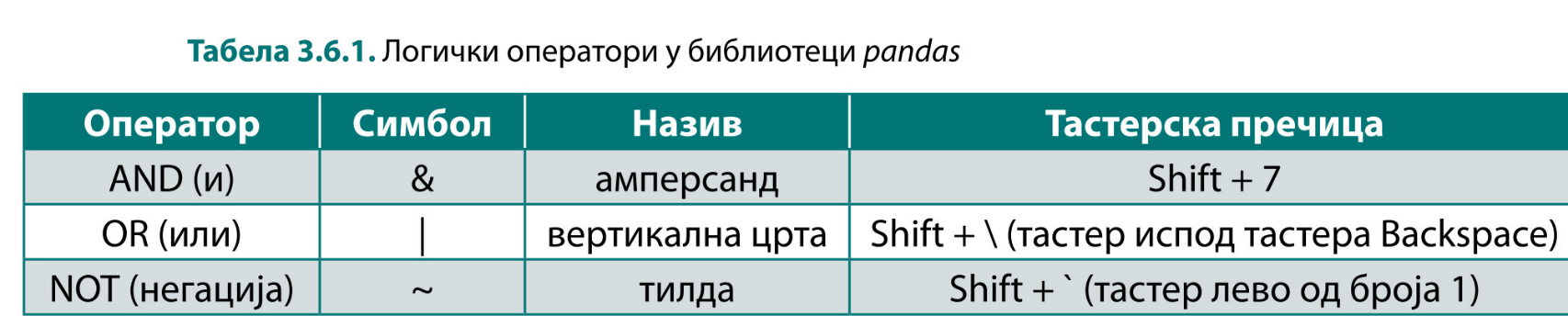
**

1. За постављање **сложених критеријума** користе се **логички оператори**.



На пример, ако хоћеш да издвојиш из табеле само **девојчице које су урадиле мање од 20 чучњева**, критеријуме ћеш издвојити у заграде а између њих ћеш унети **логички оператор.** За њега се у библиотеци **pandas** користи симбол

1. **AND (и).** **& (амперсанд).**
2. За логичке операторе **OR (или)** **| (вертикална црта)**
3. **NOT** (оператор негације) у библиотеци *pandas* користе се симболи и **~ (тилда)**, као што је приказано у табели 3.6.1.



Код филтрирања је уочљива још једна предност *Jupyter-a* над *Ехсе1-ом.* У оба програма можеш наставити да примењујеш филтере на филтрирану табелу. Ал и у *Jupyter-y* можеш добијене међурезултате чувати у посебним променљивама.

