# Променљиве

* Променљива је један од основних појмова у програмирању.
* **Променљива** је име које се даје месту у меморији рачунара у коме се могу чувати подаци.

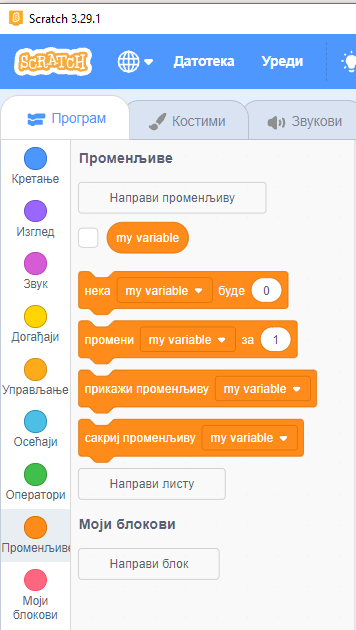
**Подаци** могу бити различитог типа

* бројчани,
* текстуални и
* други).

Може чувати **само један податак**, све док он не буде замењен другим податком.

x=0

Променљивама можеш **на почетку доделити вредности**, или можеш захтевати да се вредности унесу током извршења програма.

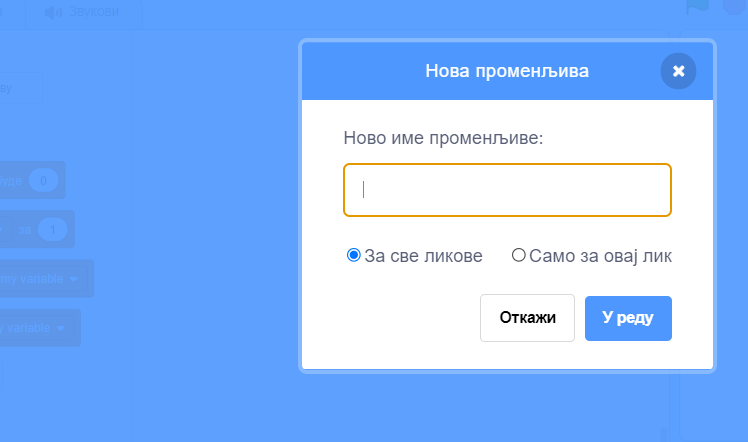
Вредност променљиве често зависи од њене **претходне вредности**. На пример, променљивој се може доделити вредност , тако да ће се њена вредност увећавати за 1

**х = x+1**

Овакав израз у математици не би имао смисла, али **променљива у математици није исто што и променљива у програмирању**.

**Прављење променљивих** у програму *Scratch* :

* 1. кликни на картицу **Програми;**
  2. кликни на ставку **Променљиве;**
  3. кликни на дугме **Направи променљиву;**



* 1. додели јој назив (у пољу **Име променљиве);**
  2. кликни **У реду.**

**Избор имена** за променљивује веома важан, иако делује једноставно. Када именујеш променљиву, мораш имати у виду следећа **три правила**.

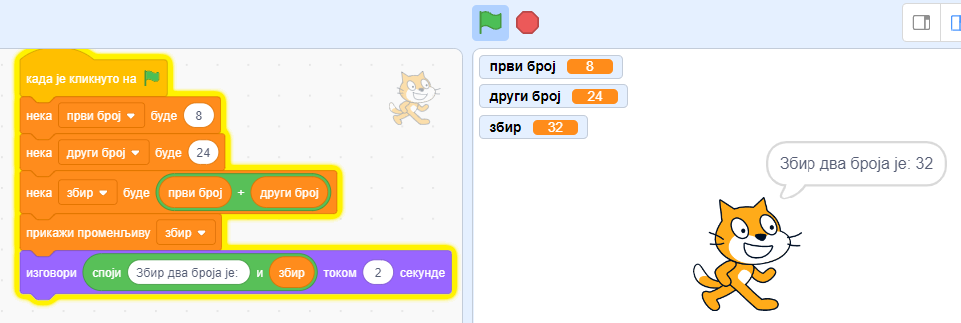
* 1. Изабери име које описује оно што ћеш у њу сачувати. На пример, ако ћеш у променљивој држати податке у виду бројева који се односе на оцене из наставних предмета, назови је ***Оцена****.*
  2. При именовању променљивих не треба бити **преопширан**. На пример, ако ће се у променљивој чувати обим правоугаоника, можеш је назвати ***Обим****.*
  3. Немој користити **посебне знакове**, зато што они могу изазвати грешке. Мислимо на знакове као што су **! $ % \*.**
  4. Немој користити **број на почетку имена**. На пример, не употребљавај име ***9Резултат****.*

### Пример 1. Програм за сабирање два броја

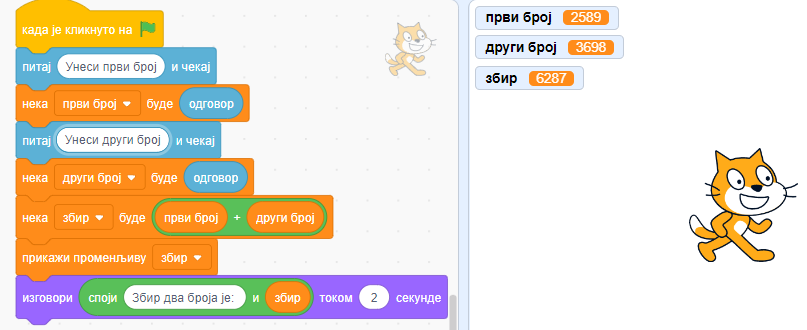
Потребно је да се креирају **три променљиве**. Назваћемо их: *Први број, Други број* и *Збир.*

Променљивој *Први број* **додељена** је вредност 8, а променљивој *Други број* 24.

Вредност променљиве *Збир* добија се на основу дефинисаног израза *(Збир = Први број+Други број).*

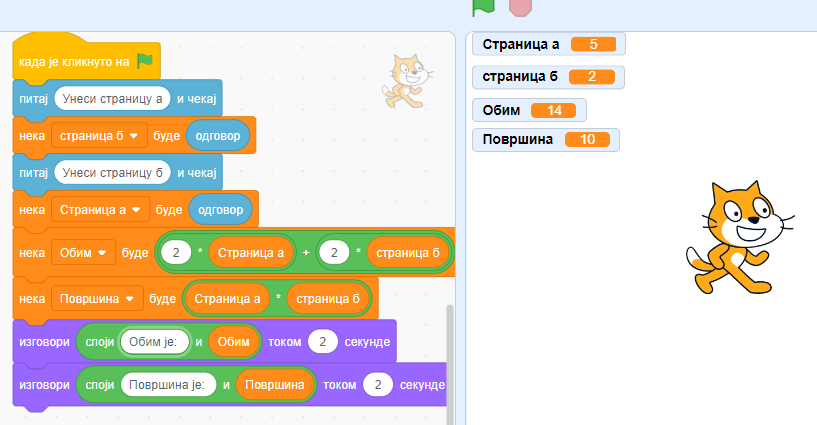


### Пример 2, сваки пут када се покрене програм, треба да тражи да унесеш вредности за променљиве *Први број* и *Други број.*

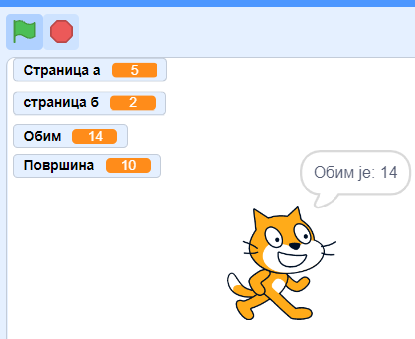
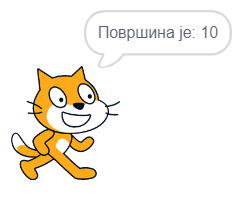


Вредности променљивама се додељују помоћу блок-наредби  Сваки следећи пут када покренеш програм, можеш променљивама доделити друге вредности и увек ћеш добити одговарајући збир.

### Пример 3. Обим и површина правоугаоника.



**четири променљиве**: *Страница а, Страница б, Обим* и *Површина*



На слици где је приказан резултат програма видиш и које вредности су додељене променљивама *Страница а* и *Страница б.* Сваки следећи пут када покренеш програм, можеш променљивама, доделити друге вредности.

### Пример 4. Израчунавање просечне вредности.

Направићемо програм који рачуна просечну температуру у токуј едногдана. Она се рачуна као збир јутарње, дневне и вечерње температуре подељен са три.

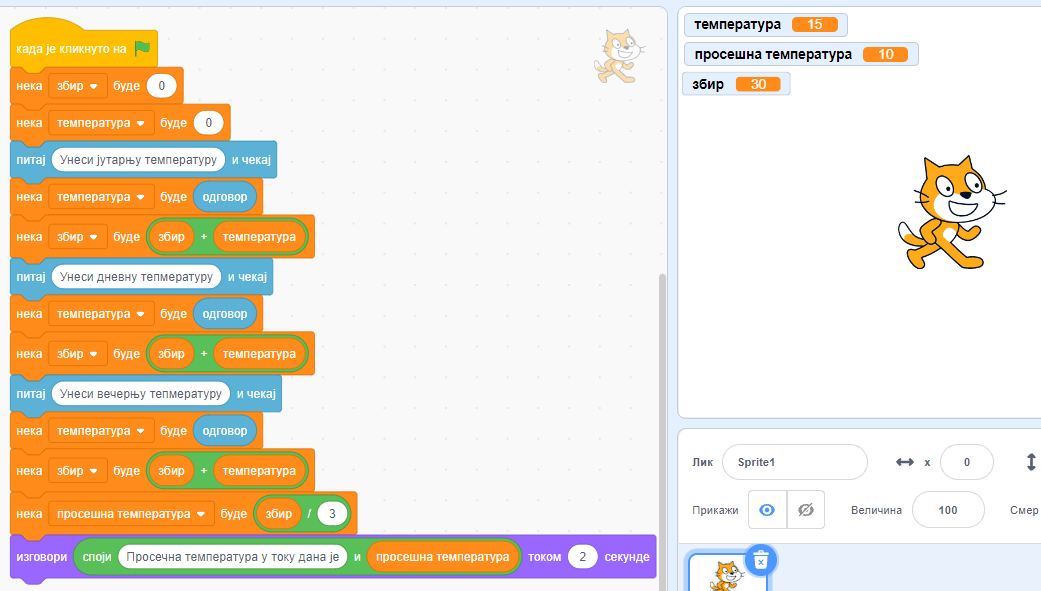
Потребноје дефинисати **три променљиве**: *Збир, Температура* и *Просечна температура.*

Променљивама ***Збир* и *Температура*** се на почетку програма увек додељују **вредности 0**.

Блок наредбом „нека **Збир буде Збир + Температура**" вредност **променљиве *Збир* се повећава сваки пут за вредност која је додељена променљивој *Температура*** (кроз унос јутарње, дневне и вечерње температуре). Овакви изрази су веома **чести у програмирању (али не и у математици**).

На крају програма, променљива *Просечна температура* узима вредност променљиве *Збир* подељене са 3.

Уношењем јутарње, дневне и вечерње температуре програм ће израчунати просечну температуру током дана.



### Дом 1. множење, дељење и разлика два броја

### Дом 2. програм за израчунавање просечне температуре током пет дана. Књига стр.160

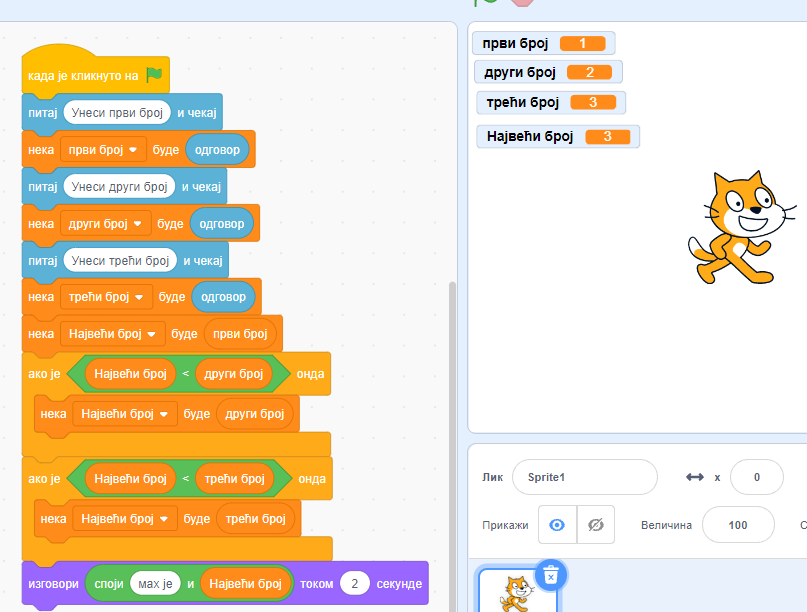
### Пример 4. програм тражи највећи од три броја.

**променљиве**: *Први број, Дру!и број, Трећи број* и *Највећи број.*

У структури програма на почетку се уносе бројеви свакој од прве три променљиве доделиш неки број.

Након тога **ћемо први број прогласити највећим променљивој** *Највећи број* се додељује вредност првог броја.

Сада следи поређење. Помоћу **наредбе АКО ЈЕ ОНДА** испитује се да ли је тренутно проглашени највећи број мањи од другог броја. Уколико јесте, променљивој *Највећи број* додељује се нова вредност, вредност другог броја. Уколико није, прелази се на наредну АКО ЈЕ ОНДА наредбу. Ту се испитује да ли је тренутно највећи број мањи од трећег броја. Уколико јесте, променљивој *Највећи број* додељује се вредност трећег броја. Уколико није, вредност променљиве *Највећи број*

****

# Заштита програма од грешака приликом уношења вредности

Претпостављамо да си све програме које си направио до сада и тестирао да видиш како функционишу. Том приликом си уносио вредности које си сам предвидео да треба да буду унете у дата поља. На пример, у програмима за рачунање помоћу бројева, уносио си цифре; када је лик требало нешто да каже, уносио си текст итд. Међутим, програме ђеш ретко када правити за себе. Обично ће они бити намењени другим корисницима.

На следећој слици је дат пример програма за рачунање обима троугла. Ако корисник унесе 56 за вредност променљиве *а,* израчуната вредност променљиве *Обим* биће 168 (3\*56). Дакле, програм савршено функционише.

Међутим, шта ће се десити ако корисник унесе вредност која није предвиђена за ту променљиву? На пример, низ слова? Програм ће изненада престати да ради. Да би се то спречило, искусни програмери велику пажњу посвећују заштити програма од уноса нелогичних вредности. Ако пажљивије погледаш претходну слику, уочићеш услов којим програм проверава да ли је *Обим* већи од 0. Ако корисник унесе вредност која не одговара датом пољу, добиће обавештење да је погрешио.

