

Podmienky

Poznámky k téme

Doteraz sa každý program vykonal vždy rovnako – riadok po riadku. Skutočné programy sa však **rozhodujú**: ak je teplota pod nulou, upozornia na poľadovicu; ak je heslo správne, pustia ťa ďalej. Na rozhodovanie slúži príkaz `if` a jeho rozšírenia `else` a `elif`.

Čo sa naučíš

- zapísať podmienku `if` so správnym odsadením,
- vybrať jednu z dvoch možností pomocou `if – else`,
- vybrať jednu z viacerých možností pomocou `elif` a zvoliť správne poradie podmienok,
- vnárať podmienky a skladať ich pomocou `and`, `or`, `not`.

1 Príkaz if

```
teplota = int(input("Teplota: "))
if teplota < 0:
    print("Pozor, mrzne!")
    print("Obleč sa teplo.")
print("Pekný deň.")
```

Výstup:

```
Teplota: -3
Pozor, mrzne!
Obleč sa teplo.
Pekný deň.
```

Ako príkaz `if` funguje:

1. Za slovom `if` je **podmienka** – výraz, ktorý je pravdivý (`True`) alebo nepravdivý (`False`).
2. Na konci riadku je **dvojbodka** `:`.
3. Pod ním nasleduje **blok** – riadky odsadené o 4 medzery. Vykonajú sa iba vtedy, keď podmienka platí.
4. Prvý riadok, ktorý už nie je odsadený (`print("Pekný deň.")`), do bloku nepatrí – vykoná sa vždy.

Pre teplotu 5 by sa vypísalo iba „Pekný deň.“

1.1 Odsadenie

V Pythone určuje **odsadenie** (medzery na začiatku riadku), ktoré príkazy patria do bloku. Iné jazyky na to používajú zložené zátvorky, Python nie. Pravidlá:

- všetky riadky jedného bloku musia mať **rovnaké** odsadenie,
- zvykom sú **4 medzery** (editor ich zvyčajne vloží klávesom Tab),
- v jednom programe nemiešaj medzery s tabulátormi.

```
vek = 20
if vek >= 18:
    print("Vstup povolený")
```

Výstup (program skončí chybou):

```
File "program.py", line 3
    print("Vstup povolený")
    ^^^^^
IndentationError: expected an indented block after 'if' statement on line 2
```

Blok za dvojbodkou nesmie byť prázdny. Ak v ňom zatiaľ nič nechceš mať, napíš príkaz `pass` (nerobí nič).

2 if – else

Vetva `else` („inak“) sa vykoná vtedy, keď podmienka **neplatí**. Vždy sa teda vykoná práve jedna z dvoch vetiev:

```
n = int(input("Zadaj číslo: "))
if n % 2 == 0:
    print(n, "je párne")
else:
    print(n, "je nepárne")
```

Výstup:

```
Zadaj číslo: 7
7 je nepárne
```

`else` nemá vlastnú podmienku a píše sa na rovnakej úrovni odsadenia ako `if`, ku ktorému patrí.

3 elif: viac možností

Keď je možností viac, použijeme `elif` (skratka *e*lse *i*f, „inak ak“). Python skúša podmienky **zhora nadol** a vykoná **iba prvú vetvu**, ktorej podmienka platí. Ostatné preskočí. Ak neplatí žiadna, vykoná sa `else` (ak je uvedený).

Príklad: známka podľa percent – aspoň 90 % je 1, aspoň 75 % je 2, aspoň 50 % je 3, aspoň 30 % je 4, inak 5.

```
percenta = int(input("Percentá: "))
if percenta >= 90:
    znamka = 1
elif percenta >= 75:
    znamka = 2
elif percenta >= 50:
    znamka = 3
elif percenta >= 30:
    znamka = 4
else:
    znamka = 5
print("Známka:", znamka)
```

Výstup:

```
Percentá: 68
Známka: 3
```

Pri 68 % podmienky `>= 90` ani `>= 75` neplatia, ale `>= 50` áno – vypíše sa 3 a zvyšok sa preskočí. Vďaka tomu netreba písať `percenta >= 50 and percenta < 75`; horná hranica je zaručená tým, že predchádzajúce vetvy neplatili.

3.1 Na poradí záleží

Ak by sme podmienky napísali v opačnom poradí, program by bol zlý:

```
percenta = int(input("Percentá: "))
if percenta >= 30:
    print("Známka 4")
elif percenta >= 90:
    print("Známka 1")
else:
    print("Známka 5")
```

Výstup:

```
Percentá: 95
Známka 4
```

Pre 95 % platí hneď prvá podmienka (`95 >= 30`), takže sa vypíše 4 a na `>= 90` sa nikdy nedostane. **Prísnejšie podmienky patria vyššie.**

3.2 elif verzus samostatné if

Niekoľko samostatných `if` sa vyhodnotí **všetky** – každý zvlášť. Reťaz `if – elif – else` vyberie **najviac jednu** vetvu. Porovnaj:

```
n = int(input())
if n > 0:
    print("kladné")
if n < 0:
    print("záporné")
else:
    print("nula")
```

Výstup:

```
5
kladné
nula
```

Pre 5 sa vypíše „kladné“ a hneď nato aj „nula“: `else` patrí iba k druhému `if` (`n < 0` neplatí). Správne je druhé `if` nahradiť `elif` – potom tvoria všetky tri vetvy jeden celok.

4 Zložené podmienky

Podmienky môžeš spájať logickými operátormi `and`, `or`, `not` (téma Operátory):

```
vek = int(input("Vek: "))
suhlas = input("Súhlas rodičov (ano/nie): ")
if vek >= 18 or (vek >= 15 and suhlas == "ano"):
    print("Môžeš sa prihlásiť.")
else:
    print("Prihlásenie nie je možné.")
```

Výstup:

```
Vek: 16
Súhlas rodičov (ano/nie): ano
Môžeš sa prihlásiť.
```

Užitočný je aj operátor `in`, ktorý zistí, či je hodnota medzi vymenovanými možnosťami alebo či je znak v texte:

```
pismeno = input("Písmeno: ")
if pismeno in "aeiouy":
    print("samohláska")
else:
    print("nie je samohláska")
```

Výstup:

```
Písmeno: a
samohláska
```

5 Vnorené podmienky

V bloku podmienky môže byť ďalšia podmienka – hovoríme o **vnorení**. Vnútoraná podmienka sa vyhodnotí iba vtedy, keď platí vonkajšia:

```
n = int(input("Číslo: "))
if n > 0:
    if n % 2 == 0:
        print("kladné párne")
    else:
        print("kladné nepárne")
else:
    print("nie je kladné")
```

Výstup:

```
Číslo: 12
kladné párne
```

Vnútorný blok je odsadený o ďalšie 4 medzery. Vnorenie sa často dá nahradiť podmienkou s `and` – vyber zápis, ktorý sa lepšie číta. Príliš hlboké vnorenie (viac ako 2–3 úrovne) je signál, že by program mal byť zapísaný jednoduchšie.

6 Pravdivosť hodnôt

V podmienke nemusí byť iba porovnanie. Python vie posúdiť pravdivosť každej hodnoty: **nepravdivé** sú `0`, `0.0`, prázdny text `""`, prázdny zoznam `[]`, `None` a `False`; všetko ostatné je pravdivé.

```
meno = input("Meno: ")
if meno:
    print("Ahoj,", meno)
else:
    print("Meno nebolo zadane.")
```

Výstup:

```
Meno:
Meno nebolo zadane.
```

Zápis `if meno:` znamená „ak meno nie je prázdne“.

7 Časté chyby

Chyba	Ako sa prejaví	Oprava
<code>if x = 5:</code>	<code>SyntaxError</code>	porovnanie je <code>==</code>
chýba dvojbodka	<code>SyntaxError: expected ':'</code>	<code>if x > 5:</code>
neodsadený blok	<code>IndentationError</code>	odsad' 4 medzerami
dva samostatné <code>if</code> namiesto <code>elif</code>	vypíše sa viac vetiev	<code>if – elif – else</code>
zlé poradie <code>elif</code>	prísnejšia vetva sa nikdy nevykoná	prísnejšie podmienky vyššie
porovnanie textu s číslom	podmienka nikdy neplatí	preveď vstup na <code>int</code>

Posledný riadok tabuľky: `input()` vracia text, takže `"18" == 18` je `False` a podmienka `if vek == 18:` by bez prevodu nikdy neplatila.

Pri porovnávaní textov pozor aj na veľké písmená: `"Áno" == "áno"` je `False`. Pomôže previesť vstup na malé písmená: `odpoved.lower() == "áno"`.

8 Riešené úlohy

8.1 Najväčšie z troch čísel

Zadanie: Načítaj tri čísla a vypíš najväčšie z nich (bez funkcie `max`).

Riešenie: Predpokladáme, že najväčšie je prvé, a postupne ho porovnáme s ďalšími. Tento postup sa neskôr použije aj na hľadanie maxima v zozname.

```
a = int(input())
b = int(input())
c = int(input())
najvacsie = a
if b > najvacsie:
    najvacsie = b
if c > najvacsie:
    najvacsie = c
print("Najväčšie:", najvacsie)
```

Výstup:

```
7
19
4
Najväčšie: 19
```

Tu sú správne dva **samostatné if**: obe porovnania sa musia vykonať.

8.2 Trojuholník

Zadanie: Načítaj dĺžky troch strán. Rozhodni, či z nich možno zostrojiť trojuholník, a ak áno, či je rovnostranný, rovnoramenný alebo rôznostranný.

Riešenie: Trojuholník existuje, ak je súčet každých dvoch strán väčší ako tretia strana (*trojuholníková nerovnosť*). Najprv skontrolujeme existenciu, až potom typ. Rovnostranný preveríme pred rovnoramenným, lebo rovnostranný trojuholník je zároveň aj rovnoramenný.

```
a = float(input("a: "))
b = float(input("b: "))
c = float(input("c: "))
if a + b > c and a + c > b and b + c > a:
    if a == b == c:
        print("rovnostranný")
    elif a == b or b == c or a == c:
        print("rovnoramenný")
    else:
        print("rôznostranný")
else:
    print("Trojuholník neexistuje.")
```

Výstup:

```
a: 5
b: 5
c: 8
rovnoramenný
```

8.3 Kvadratická rovnica

Zadanie: Načítaj koeficienty a, b, c rovnice $ax^2 + bx + c = 0$ ($a \neq 0$) a vypíš jej reálne korene.

Riešenie: Počet koreňov závisí od diskriminantu $D = b^2 - 4ac$: pre $D > 0$ sú dva korene, pre $D = 0$ jeden, pre $D < 0$ žiadny reálny koreň. Odmocninu vypočítame ako `D ** 0.5`.

```
a = float(input("a: "))
b = float(input("b: "))
c = float(input("c: "))
D = b ** 2 - 4 * a * c
if D > 0:
    x1 = (-b + D ** 0.5) / (2 * a)
    x2 = (-b - D ** 0.5) / (2 * a)
    print(f"x1 = {x1}, x2 = {x2}")
elif D == 0:
    print("x =", -b / (2 * a))
else:
    print("Rovnica nemá reálne korene.")
```

Výstup:

```
a: 1
b: -5
c: 6
x1 = 3.0, x2 = 2.0
```

8.4 Kameň, papier, nožnice

Zadanie: Načítaj svoju voľbu a voľbu počítača (`kamen` , `papier` , `noznice`) a vypíš výsledok. Kameň tupí nožnice, nožnice strihajú papier, papier zabalí kameň.

Riešenie: Najprv vyriešime remízu. Potom vymenujeme tri situácie, keď vyhráš ty; vo všetkých ostatných vyhrá počítač. V skutočnej hre by počítač volil náhodne (`random.choice` , téma Modulárny program).

```
hrac = input("Hráč: ")
pocitac = input("Počítač: ")
if hrac == pocitac:
    print("Remíza")
elif (hrac == "kamen" and pocitac == "noznice") or \
     (hrac == "noznice" and pocitac == "papier") or \
     (hrac == "papier" and pocitac == "kamen"):
    print("Vyhrávaš!")
else:
    print("Vyhráva počítač.")
```

Výstup:

```
Hráč: papier
Počítač: kamen
Vyhrávaš!
```

Dlhú podmienku sme rozdelili na viac riadkov znakom `\` na konci riadku.

8.5 Cena lístka

Zadanie: Lístok stojí 8 €. Deti do 6 rokov majú vstup zadarmo, deti do 15 rokov a ľudia od 62 rokov platia polovicu. Načítaj vek a vypíš cenu.

```
vek = int(input("Vek: "))
if vek < 6:
    cena = 0
elif vek < 15 or vek >= 62:
    cena = 4
else:
    cena = 8
print(f"Cena: {cena} €")
```

Výstup:

```
Vek: 64
Cena: 4 €
```

Zhrnutie

- `if` podmienka: + odsadený blok – blok sa vykoná, iba ak podmienka platí.
- `else:` – vykoná sa, keď podmienka neplatí; vždy sa vykoná práve jedna z vetiev.
- `elif` – ďalšia podmienka; vykoná sa iba **prvá** platná vetva, preto prísnejšie podmienky píš vyššie.
- Samostatné `if` sa vyhodnocujú všetky, reťaz `if – elif – else` vyberie jednu vetvu.
- Odsadenie (4 medzery) určuje blok; za `if`, `elif`, `else` je dvojbodka.
- `0`, `""`, `[]`, `None` a `False` sú v podmienke nepravdivé.

Úlohy na precvičenie

1. Načítaj číslo a vypíš, či je kladné, záporné alebo nula.
2. Načítaj rok a vypíš, či je priestupný – tentoraz pomocou vnorených podmienok.
3. Načítaj body z písomky (0 – 50) a vypíš známku podľa stupnice 45, 38, 25, 16 bodov.
4. Načítaj tri čísla a vypíš ich zoradené od najmenšieho.
5. Načítaj hmotnosť a výšku, vypočítaj BMI a vypíš kategóriu: pod 18,5 podváha, do 25 norma, do 30 nadváha, inak obezita.

Úlohy k tejto téme nájdeš na learnwithmente.com v časti Informatika → Python.