

NASTAVENIE PROSTREDIA KROK ZA KROKOM

⚙️ Nainštaluj si Visual Studio Code

- 🔍 **Čo to je?** Editor kódu.
- 💡 **Prečo to potrebujem?** Nie je to *nevyhnutné*, ale uľahčuje programovanie farebným zvýraznením častí kódu, automatickým dopĺňaním a zvýrazňovaním chýb.
- 🔗 **Kde to získam?** [Tu](#) 📄

⚙️ Nainštaluj si Python

- 🔍 **Čo to je?** Programovací jazyk (presnešie compiler).
- 💡 **Prečo to potrebujem?** Povie tvojmu počítaču ako vykonávať programy napísané Python syntax.
- 🔗 **Kde to získam?** [Tu](#) 📄

⚙️ Nainštaluj si Python extension (rozšírenie)

- 🔍 **Čo to je?** Dodatočné užitočné doplnky pre editor kódu.
- 💡 **Prečo to potrebujem?** Dodá pomoc počas programovania špecifickú pre programovanie v Pythone.
- 🔗 **Kde to získam?** VSCode → ctrl + shift + x → vyhľadaj 'Python Extension Pack' → 'install'

⚙️ Vytvor si GitHub účet

- 🔍 **Čo to je?** Open-source online platforma na ukladanie a zdieľanie kódu.
- 💡 **Prečo to potrebujem?** Na zálohovanie tvojho kódu, spoluprácu s ostatnými a prezentovanie projektov.
- 🔗 **Kde to získam?** [Tu](#) 🔗

⚙️ Nainštaluj si Git

- 🔍 **Čo to je?** Lokálny systém riadenia verzií.
- 💡 **Prečo to potrebujem?** Na monitorovanie zmien v tvojom kóde a nahrávanie zmien na GitHub.
- 🔗 **Kde to získam?** [Tu](#) 📄

⚙️ Nainštaluj si Git extension & prihlás sa

- 🔍 **Čo to je?** Doplnky editora kódu na prácu s Git.
- 💡 **Prečo to potrebujem?** Na používanie Git cez VSCode rozhranie namiesto terminálu.
- 🔗 **Kde to získam?** VSCode → ctrl + shift + x → vyhľadajte 'GitHub Pull Requests' → inštalovať; prihláste sa do vášho GitHub účtu vo VSCode → Accounts (ľavý dolný roh)

⚙️ Skonfiguruj si Git vo VSCode

- 🔍 **Čo to je?** Povieš tvojmu VSCode, aby komunikoval s *tvojím* GitHubom.
- 💡 **Prečo to potrebujem?** Na identifikáciu, koho zmeny sa ukladajú na GitHub.
- 🔗 **Ako na to?** VSCode → ctrl + ` → git config --global user.name "[tvoje používateľské meno]" → enter → git config --global user.email "[tvoj email]" → enter

Teraz môžeš začať kódovať!

SPUSTENIE PROJEKTU



Vytvor svoj prvý projekt

- 1 Vo VSCode choď do Explorer (ikona s dokumentami alebo ctrl + shift + E)
↓
- 2 Stlač 'Open Folder' a naviguj do priečinka tvojho projektu (alebo si vytvor nový priečinok)
↓
- 3 Vyber si požadovaný priečinok a stlač 'Open'
↓
- 4 V Exploreri naviguj k názvu priečinka na ľavej strane a stlač 'New File'
↓
- 5 Pomenuj svoj súbor a pridaj koncovku .py, aby VSCode vedel, že toto je súbor napísaný v Pythone.



Nahraj svoj projekt online

- 1 Vo VSCode choď do Source Control (ikona vetvy alebo ctrl + shift + G)
↓
- 2 Stlač 'Initialize Repository'
↓
- 3 Do textového poľa na ľavej strane napíš krátku správu opisujúcu tvoj súbor a stlač 'Commit'
↓
- 4 stlač 'Publish Branch'
↓
- 5 V dropdown menu vyber 'Publish to GitHub public repository'
↓
- 6 Skontroluj na [GitHub](#), že existuje repository s názvom tvojho projektu obsahujúci tvoj súbor



Navrhni zmeny v cudzích projektoch

- 1 Na GitHube choď do repository, ktorú plánuješ upraviť, stlač zelené tlačidlo 'Code' a skopíruj URL
↓
- 2 Vo VSCode choď do Source Control (ctrl + shift + G) a stlač 'Clone Repository'
↓
- 3 Vlož skopírovanú URL do zobrazeného textového poľa a stlač enter
↓
- 4 Naviguj, kde chcete uložiť repository a stlač 'Select as Repository Destination'
↓
- 5 Source Control → 'More Actions' (3 bodky vedľa 'CHANGES') → 'Branch' → 'Create Branch'
↓
- 6 Pomenuj svoju vetvu a stlač enter
↓
- 7 Urob zmeny v kóde
↓
- 8 V GitHub tabe na VSCode stlač 'Create Pull Request' (vedľa 'PULL REQUESTS') a napíš správu

**Inkorporuj návrhy zmien od ostatnýchs**

- 1 Vo svojom VSCode stlač ikonu 'sync changes' (dolná ľavá lišta)
↓
- 2 V GitHub Pull Requests tabe na VSCode nájdeš svoje nevyriešené pull requesty
↓
- 3 Vyber si pull request a stlač 'Open Changes' (ikona vedľa názvu requestu)
↓
- 4 Pozri si zmeny a prípadne pridaj komentáre stlačením '+' na zmenených riadkoch
↓
- 5 Stlač názov requestu na otvorenie tabu s detailmi requestu
↓
- 6 Vyber 'Approve' na zapracovanie zmeny, 'Comment' na okomentovanie zmeny, alebo 'Request Changes' na vyžiadanie zmien k navrhovanej zmene

💡 Primitívne dátové typy

NÁZOV	UKLADÁ	PRÍKLADY
int (celé číslo)	celočíselnú hodnotu	3, 7, 42
float (desatinné číslo)	desatinnú hodnotu	3.14, 2.0
str (reťazec znakov)	text	'Sherlock', "Holmes"
bool (logická hodnota)	True alebo False hodnoty	True, False

Môžeš vytlačiť dátový typ pre `x` spustením `print(type(x))`. Nauč sa viac o iných dátových typoch [tu](#) .

💡 Operátory

PRIRADŔOVACIE OPERÁTORY

OPERÁTOR	FUNKCIA	PRÍKLAD	EKVIVALENT
=	priradí hodnotu k názvu	<code>x = 2</code>	
+=	zvýši priradenú hodnotu o novú hodnotu	<code>x += 3</code>	<code>x = x + 3</code>
-=	zníži priradenú hodnotu o novú hodnotu	<code>x -= 3</code>	<code>x = x - 3</code>
=	umocní priradenú hodnotu na novú hodnotu	<code>x **= 3</code>	<code>x = x3</code>

ARITMETICKÉ OPERÁTORY


OPERÁTOR	VRACIA	PRÍKLAD
+	súčet dvoch hodnôt	<code>x + y</code>
-	rozdiel dvoch hodnôt	<code>x - y - z</code>
*	súčin dvoch hodnôt	<code>x * y</code>
/	podiel dvoch hodnôt	<code>x / y</code>
**	umocnenie (mocnina) dvoch hodnôt	<code>x ** y</code>
%	zvyšok po delení (modulo)	<code>x % y</code>

LOGICKÉ OPERÁTORY


OPERÁTOR	VRACIA	PRÍKLAD
and (a)	True ak sú obe podmienky pravdivé	<code>x > 5 and y < 10</code>
or (alebo)	True ak je aspoň jedna podmienka pravdivá	<code>x > 5 or y < 10</code>
not (nie)	True ak je podmienka nepravdivá (negácia)	<code>not x > 5</code>

POROVNÁVACIE OPERÁTORY

OPERÁTOR	VRACIA	PRÍKLAD
==	True ak sú hodnoty rovnaké	<code>x == 5</code>
!=	True ak sa hodnoty nerovnajú	<code>x != 5</code>
>	True ak je ľavá hodnota väčšia	<code>x > 5</code>
<	True ak je ľavá hodnota menšia	<code>x < 5</code>
>=	True ak je ľavá hodnota väčšia alebo rovná	<code>x >= 5</code>
<=	True ak je ľavá hodnota menšia alebo rovná	<code>x <= 5</code>

Zisti viac o ďalších operátoroch [tu](#) .

**Všeobecné tipy**

- 1 **Veľa problémov sa dá vyriešiť pozorným čítaním upozornení a manuálov.** Keď sa počítač snaží komunikovať, venuj chvíľu prečítaniu toho, čo hovorí—zvyčajne sa snaží pomôcť.
- 2 **Väčšina programátorov googli (a používa AI!) takmer neustále.** Čokoľvek ťa trápí zrejme už trápilo niekoho pred tebou; pozri, či ti niekoho rada náhodou nepomôže.
- 3 Vďaka Jupyter extension, ktoré si si nainštalovala ako súčasť Python Extension Pack môžeš napísať **`#%%`**, čím rozdelíš svoj kód na ‘bunky’, ktoré môžeš spustiť osobitne (netreba vytvárať nový súbor pre každú úlohu).
- 4 **Vždy dokumentuj svoj kód.** Je to jednoduché: len napíš **`##`** vedľa konkrétnych riadkov kódu a napíš stručný opis toho, čo príslušný riadok robí. (Môžeš dokumentovať aj otázky či obmedzenia.)
- 5 **Používaj print statementy.** Do zátvoriek `print()` napíš obsah a nájdi ho vytlačený v termináli. Zisti ako používať `print()` v rôznych situáciách [tu](#) .