

Želva – pracovní list

Želví grafika v prostředí Python

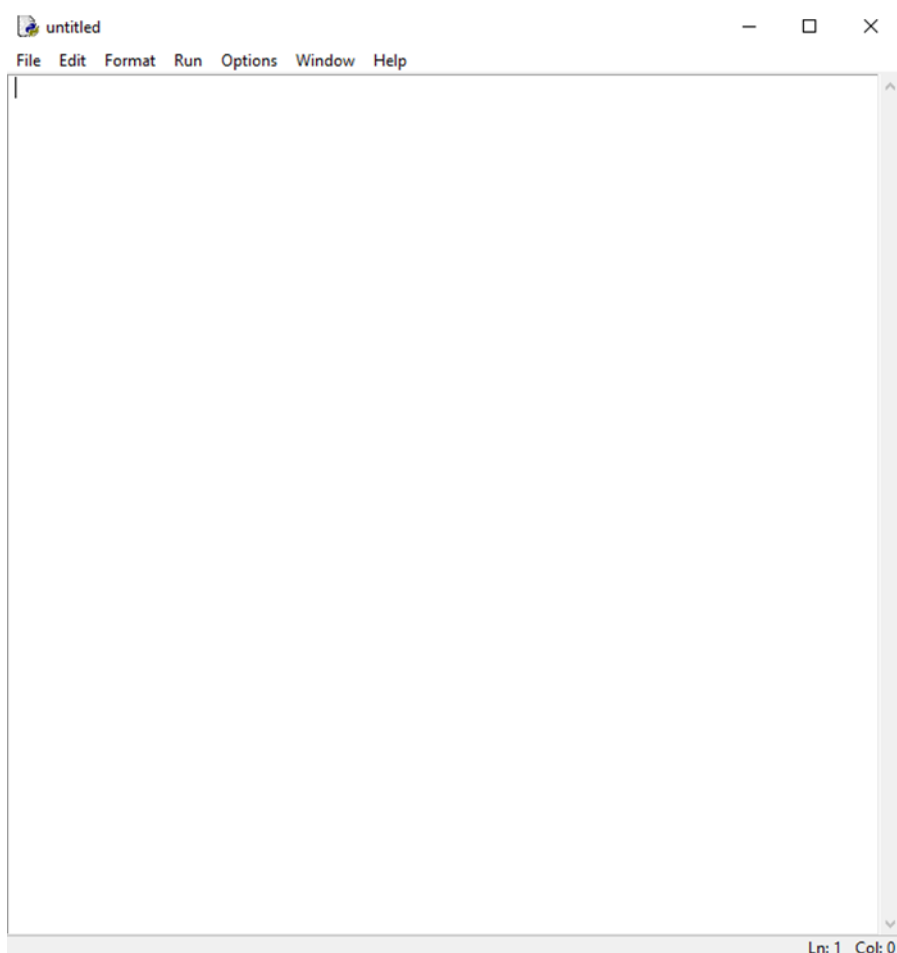
Začínáme

V dnešní hodině si budeme hrát na umělce. A jelikož se jedná o hodinu informatiky, nebudeme kreslit na papír. Nebudeme však ani kreslit v programu Malování, který jistě všichni znáte. Budeme se věnovat takzvané želví grafice.

Co budeme potřebovat?

Důležité je mít nainstalované vývojové prostředí. Toto prostředí je nainstalované na všech školních počítačích.

Takto prostředí vypadá:



Zdroj: Kolektiv autorů

Soubor si uložíme do naší složky a nazveme ho zelva.py.

Začínáme programovat

Než se pustíme do programování, musíme přidat knihovnu, která je nazvaná turtle – to provedeme takto: `import turtle`.

Následně si definujeme plátno, na které budeme kreslit a naši želvu.

```
platno = turtle.Screen()
```

```
platno.bgcolor("black")
```

```
platno.title("Želva")
```

```
zelva = turtle.Turtle()
```

```
loadWindow = turtle.Screen()
```

```
zelva.pencolor('red')
```

Kreslíme

Pro kreslení máme několik základních příkazů.

`Forward (x)` – posun želvy dopředu. `X` nám značí, o kolik se želva posune.

`Right (x)` – zatoč doprava. `X` nám značí, o kolik stupňů se želva otočí.

`Left (x)` – zatoč doleva. `X` nám značí, o kolik stupňů se želva otočí.

Samozřejmě musíme říci, kdo tento povel vykoná. V našem případě je to želva.

Ukázka, jak nakreslit čáru – `zelva.forward(100)`



Zdroj: Kolektiv autorů

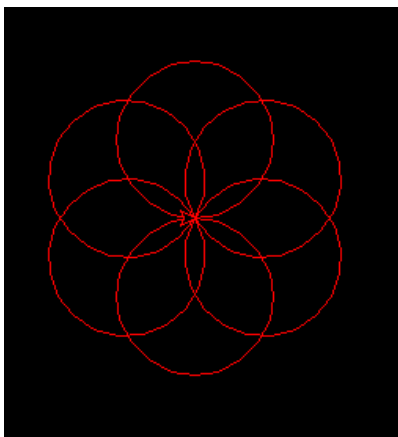
Zadání

Nakreslete pomocí želví grafiky a vložte `printscreen` obrazovky do tohoto dokumentu. Využijte cykly:

Čtverec

Kružnice

Vytvořte kružnice, které se takto překrývají:



Zdroj: Kolektiv autorů

Závěrečná práce

Vytvořte ve skupinkách svůj vlastní obrazec. Rozmyslete si, jak váš program bude fungovat, a jeho funkci nám poté popíšete a výsledný obrazec předvedete.

Užitečné příkazy najdete na adrese:

<https://docs.python.org/3.3/library/turtle.html?highlight=turtle>

