

```
from pathlib import
```

# Path

## Tvorba cest

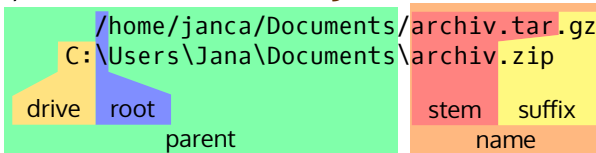
```
p = Path('/home/janca/Documents/archiv.tar.gz')
p = Path('/', 'home', 'janca/Documents', 'archiv.tar.gz')
p = Path('/home') / 'janca' / Path('Documents') / 'archiv.tar.gz'
p = Path('/home').joinpath('janca', 'Documents/archiv.tar.gz')
```

## Zvláštní cesty

```
Path.cwd()   Aktuální adresář
Path.home()  Domovský adresář
```

## Části cest

```
p.name == 'archiv.tar.gz'
```



```
p.suffixes == ['.tar', '.gz']
p.parents == [Path('/home/janca/Documents'),
              Path('/home/janca'),
              Path('/home'), Path('/')]
p.with_name('zaloha.tgz') Cesta s přepsaným jménem
p.with_suffix('.foo')    Cesta s přepsanou/přidanou příponou
```

## Absolutní a relativní cesty

```
p.is_absolute()  Je p absolutní cesta?
p.relative_to(p2) Vytvoří relativní cestu vzhledem k p2
p.resolve()      Vytvoří absolutní cestu
```

## Soubory pojmenované cestou

```
p.exists()       Existuje na cestě p soubor?
p.is_dir()       Je to adresář?
p.is_file()      Je to normální datový soubor?
p.samefile(p2)  Pojmenovávají cesty p a p2 stejný soubor?
p.unlink()      Odstraní normální soubor
p.rename(p2)    Přejmenuje/přesune soubor na novou cestu
```

## Adresáře (složky)

```
p.mkdir()        Vytvoří adresář
p.rmdir()        Smaže prázdný adresář
p.iterdir()      Projde obsah adresáře – for c in p.iterdir() ...
p.glob(m)        Projde obsah adresáře podle šablony jména
```

## Čtení a zápis datových souborů

```
with p.open(encoding="utf-8") as f:
    contents = f.read()
contents = p.read_text()
contents = p.read_bytes()
p.write_text("Ahoj světe!")
p.write_bytes(b'\x00\xff')
```

## Cesty podle OS

PosixPath (Linux/macOS)  
Oddělovač: '/'  
Nemá drive  
Rozlišuje velikost písmen

WindowsPath  
Oddělovač: '\\'  
Má drive (C:, D:, ...)  
Nerozlišuje velikost

## Šablony pro glob

? Jeden znak  
\* 0 nebo více znaků  
\*\* Podadresáře  
[ab] Jeden z daných znaků