

Naključna števila

- `import random`
- `random.random()`
- Dobimo decimalno število z intervala [0.0, 1.0)

random.randint(a, b)

- Dobimo naključno celo število med **a** in **b**. Meji sta **vključeni**, torej lahko dobimo tako **a** kot tudi **b**.
- Kocka:
 - `random.randint(1, 6)`
- Ruleta (francoska):
 - `random.randint(0, 36)`
- Poker:
 - Barva:
 - `random.randint(1, 4)`
 - Vrednost:
 - `random.randint(2, 14)`



Zgledi

- Sestavi funkcijo, ki vrne par (barva, vrednost) , ki pomeni neko naključno karto
- Barva: 'srce', 'pik', 'kara', 'križ'
- Vrednost: 2, 3, ..., 10, fant, dama, kralj, as
- Ideja:
 - Generiramo dve naključni števili. Prva je barva, druga vrednost.

```

import random
def karta() :
    ''' vrne par (barva, vrednost) ki predstavlja
        neko naključno karto. Oba elementa sta niza.
    '''
    barvaStev = random.randint(1,4)
    if barvaStev == 1 : barva = 'srce'
    if barvaStev == 2 : barva = 'pik'
    if barvaStev == 3 : barva = 'kara'
    if barvaStev == 4 : barva = 'križ'

    vrednostStev = random.randint(2, 14)
    if vrednostStev < 11 : # od 2 - 10
        vrednost = str(vrednostStev) # kot niz '2', ..., '10'
    elif vrednostStev == 11 : # fant
        vrednost = 'fant'
    elif vrednostStev == 12 :
        vrednost = 'dama'
    elif vrednostStev == 13 :
        vrednost = 'kralj'
    else :
        vrednost = 'as'

    return (barva, vrednost)

```

```

>>> karta()
('kara', '2')
>>> karta()
('pik', '6')
>>> karta()
('srce', 'kralj')
>>> karta()
('križ', '10')
>>> karta()
('kara', '7')

```

Če smo že boljši programerji ...

```
import random
def karta() :
    ''' vrne neko naključno karto v obliki
        ('srce', '4')
        ('kara', 'kralj')
    '''
    barve = ['srce', 'pik', 'križ', 'kara'] # seznam možnih barv
    barva = random.randint(0,3) # indeks ustrezne barve
    barvaKarte = barve[barva]

    vrednost = random.randint(2, 14)
    opisi = ['fant', 'kraljica', 'kralj', 'as']
    if vrednost > 10 :
        vrednostKarte = opisi[vrednost - 11]
    else :
        vrednostKarte = str(vrednost)

    return (barvaKarte, vrednostKarte)
```

Z uporabo choice

```
def karta() :  
    ''' vrne neko naključno karto v obliki  
        ('srce', '4')  
        ('kara', 'kralj')  
    '''  
    barve = ['srce', 'pik', 'križ', 'kara'] # seznam možnih barv  
    barva = random.randint(0,3) # indeks ustrezne barve  
    barvaKarte = random.choice(barve)  
  
    vrednosti = list(range(2,10)) + ['fant', 'kraljica',  
                                     'kralj', 'as']  
    vrednostKarte = str(random.choice(vrednosti))  
  
    return (barvaKarte, vrednostKarte)
```

Pari / nabori

- Kot rezultat dobimo npr. ('srce' , '9')
- Kaj je (a , b) ?
- To je par, oziroma nabor (z dvema vrednostima)
- Nabori - enako kot sezname, le da jih ne moremo spreminjati po komponentah
- Uvod v programiranje:
 - Vaja2 (Izračunan kot naj bo v stopinjah in minutah)
- $x = (2, 5)$
- $x[1]$ # to je 5!
- Razpakiranje:
 - prva, druga = (3, 5)

Druge karte, druga imena

- Nemčija: želod, list, srce, kraguljček
- Italija: kij, sablja, kupa, denar
- Švica: želod, ščit, roža, kragulj
- Lokalna poimenovanja
- Preferans: od 7 do As



Posplošena funkcija

```
def karta (seznamBarv, seznamVrednosti)
```

- Ideja 1:

- Generiramo indeks barve in indeks vrednosti
- Potem le vzamemo iz seznama
- Indeks barve: med 0 in dolžina seznama - 1
- `random.randint(0, len(seznamBarv) - 1)`

- Ideja 2:

- S choice vzamemo iz seznama
- `random.choice(seznamBarv)`

Privzete vrednosti

- Moti nas, da moramo pri uporabi "običajnih" kart ob klicu vnašati tako dolga seznama.
- Funkciji zato določimo **privzeto** vrednost parametrov
- `def mojaFun(par1 = 10, par2 = 5) :`
- Če kličemo:
 - `mojaFun()`
 - je to enako, kot `mojaFun(10, 5)`
 - torej `par1` je 10 in `par2` je 5
 - `mojaFun(2, 7)`
 - `par1` je 2 in `par2` je 7
 - `mojaFun(3)`
 - enako kot `mojaFun(3, 5)`

Poimenovane vrednosti

- Keyword arguments
- Če pa vemo, kako so poimenovani parametri v funkciji, lahko to izrabimo tudi pri klicu
 - `def mojaFun(par1, par2)`
- Ni pomemben vrstni red
 - `mojaFun(par2 = 2, par1 = 7)`
 - par1 je 7 in par2 je 2
- Lahko najprej navedemo nekaj "navadnih" parametrov
 - `mojaFun(7, par2 = 2)`
 - par1 je 7 in par2 je 2
- Ampak po uporabi poimenovanega parametra ne moremo več uporabiti nepoimenovanih
 - `mojaFun(par2 = 3, 5)`
 - `SyntaxError: non-keyword arg after keyword arg`
- Kje smo to srečali?
 - `print('bla', 'ble', 'blo', sep = '*', end = '//')`

```

def kartaImena(
    seznamBarv = ['srce', 'križ', 'kara', 'pik'],
    seznamVrednosti = ['2', '3', '4', '5', '6', '7', '8', '9',
                       '10', 'fant', 'dama', 'kralj', 'as']) :
    ''' vrne par (barva, vrednost) ki predstavlja
        neko naključno karto. Oba elementa sta niza.
    '''
    barvaStev = random.randint(0, len(seznamBarv) - 1) # nakl. indeks
    barva = seznamBarv[barvaStev] # poberemo ustrezno barvo
    vrednostStev = random.randint(0, len(seznamVrednosti) - 1)
    vrednost = seznamVrednosti[vrednostStev]
    return (barva, vrednost)

```

```

>>> kartaImena()
('kara', '4')
>>> kartaImena()
('srce', 'dama')
>>> kartaImena(['kij', 'sablja', 'kupa', 'denar'])
('denar', '2')
>>> kartaImena(['kij', 'sablja', 'kupa', 'denar'])
('denar', 'as')
>>> kartaImena(['rdeča', 'črna'], [1, 2, 3, 4, 5, 6])
('črna', 1)
>>> kartaImena(['rdeča', 'črna'], [1, 2, 3, 4, 5, 6])
('rdeča', 3)

```

Seveda bi funkcijo lahko napisali tudi ...

```
def kartaImena2(seznamBarv = ['srce', 'križ', 'kara', 'pik'],
                seznamVrednosti = ['2', '3', '4', '5', '6', '7', '8', '9',
                                     '10', 'fant', 'dama', 'kralj', 'as']) :
    ''' vrne par (barva, vrednost) ki predstavlja
        neko naključno karto.
    '''
    barva = random.choice(seznamBarv) # ustrezna barva
    vrednost = random.choice(seznamVrednosti)
    return (barva, vrednost)
```

Funkcije v random

- V random je cel kup uporabnih funkcij
- `random.random()`
 - Naključno decimalno število med 0 in 1 /z 0 in brez 1/
- `random.randint(a, b)`
 - Naključno celo število med a in b (vključno z a in b)
- `random.uniform(a, b)`
 - Naključno decimalno število med a in b
- `random.choice(sez)`
 - Vrne naključni element seznama sez
- `random.shuffle(sez)`
 - Premeša seznam sez. Seznam sez je potem spremenjen!
- `random.sample(vzorec, k)`
 - Vrne seznam k različnih naključnih elementov seznama vzorec

Nekaj zgledov

- Izračunaj Pi s pomočjo metode MonteCarlo
 - Če naključno generiramo točke (x,y) , kjer velja $0 \leq x \leq 1$ in $0 \leq y \leq 1$, je razmerje med točkami, ki so znotraj kroga s središčem v $(0,0)$ in s polmerom 1 in vsemi točkami, pri dovolj velikem številu poskusov $\pi/4$
- Preveri kvaliteto računalniške kocke
 - Denimo, da n-krat vržemo računalniško kocko in štejemo, kolikokrat je padlo 5 pik. Če je n velik, mora biti (število padlih 5) / n približno $1/6$.
- Napolni seznam celih števil z naključnimi vrednostmi tako, da bo urejen nepadajoče
 - A) Če je dovoljeno na koncu narediti `sort()`
 - B) Če mora seznam sproti nastajati urejeno!