

## Programátor www aplikací v Pythonu

### Okruh 1 - Základy

Každý účastník dostane 3 otázky.

#### Základní konstrukce jazyka Python

##### Proměnné a datové typy

- proměnné jsou reference (odkazy) na objekty uložené v paměti počítače (zásobník/halda)
- proměnné uchovávají adresu na svá data
- python je dynamicky typovaný jazyk – u proměnné nemusíme deklarovat typ
- typy: bool, int, float, str, (bajt, seznamy, n-tice, množiny, slovníky)

##### Parsování

- je převod z textové podoby na jiný typ

##### Podmínky

- slouží k větvení programu
- vrací True / False
- klíčová slova: if – elif – else
- alternativa: match, case, case\_ / match s porovnáváním více hodnot case 1 | 2 | 3:
- ternární výraz: hodnota1 if (výraz) else hodnota2 [jednořádkové zkrácení zápisu podmínky]

##### Operátory

- ==, >, <, >=, <=, !=, not
- and, or (operátory pro složená podmínky)

##### Cykly

- while, for
- range() – vrací vygenerovaná čísla (začátek, konec, krok)
- vnořování cyklů pro výpus tabulky
- vybírání (dle indexů): [index]
- výřez (dle indexů): [index\_od : index\_do : krok]
- pass, break, continue, return

##### Seznamy

- uclavává větší množství proměnných ke kterým přistupuje pomocí indexů
- deklaruje se [], nebo list()

##### Metody pro práci se seznamy

- append(), extend(), remove(), count() [přidat nakonec, připojit seznam, odebrat položku, počet výskytů]
- pop(), clear(), reverse(), short() [poslední pryč, všechny pryč, převrátit, seřadit]
- index(), insert() [index prvního výskytu, vložení položky na danou pozici]
- globální funkce: len(), min(), max(), sum(), sorted() [počet, největší, nejmenší, součet]
- globální funkce: sorted(), del() [nový seřazený seznam, smaže položku dle indexu]
- globální funkce: all(), any() [True/False: když splňují podmínku všechny, když alespoň jeden]

##### Metody pro práci s řetězcí

- startswith(), endswith(), a operátor in [hledání a porovnání]
- upper(), lower(), capitalize(), a title() – [změna vzhledu]
- strip(), lstrip(), a rstrip() – [oříznutí bílých znaků]
- replace() – [nahrazení textu]

- ljust(), rjust(), center() – [vlození bílých znaků, např. prázdná místa doplňující počet znaků řetězce]
- len() [globální funkce k zjištění délky řetězce]
- metody is\*() [podmínka, vrací True/False]
- isalnum(), isalpha(), isdigit(), isdecimal(), islower(), isupper(), isnumeric(), isspace(), istitle()
- speciální znaky: \n, \t, \a, "" [nový řádek, tabulátor, zpětné lomítko, zvonek, trojitě uvozovky]
- speciální znaky: \\, \", \', r'str', [zpětné lomítko, uvozovky, apostrof, zrušení funkce zpětného lomítka]
- ASCII - nahrazeno UTF-8, výhoda je že znaky jsou za sebou podle abecedy [ord(str), chr(int)]

##### Ošetření chyb

- try, except (except as), else, finally
- Value Error, SyntaxError, ZeroDivisionError, TypeError, ...

##### Knihovny

- import: import nazev\_modulu / volání: nazev\_modulu.nazev\_funkce()
- import: from nazev\_modulu import nazev\_funkce / volání: nazev\_funkce()
- import všeho: from nazev\_modulu import \*

##### Matematické funkce

- math.pi(), math.e() – pí, Eulerovo číslo
- math.sin(), math.cos(), math.tan() / math.asin(), math.acos(), math.atan()
- math.degrees() – radiány na stupně
- math.ceil(), math.floor() – zaokrouhlování
- math.fabs() – absolutní hodnota desetinného čísla
- math.factorial() – faktoriál daného čísla
- math.pow(), math.sqrt() - mocnění, odmocnění
- math.hypot() – výpočet přepony pravoúhlého trojúhelníka
- math.log(), math.log10(), math.log2() – vrátí z daného čísla logaritmus daného základu

## Programátor www aplikací v Pythonu

### Okruh 2 - OOP

Každý účastník dostane 3 otázky.

#### Objektově-orientované programování v Pythonu

##### Třídy a instance

- Třída je vzor podle kterého se objekty vytváří
- Instance je objekt, který se podle třídy vytvoří
- Zapouzdření, Dědičnost a Polymorfismus

##### Atributy a metody

- Atributy jsou proměnné (odkazy na data), která jsou uvnitř instance
- Metody jsou funkce (schopnostmi objektu), ke kterým přistupujeme skrze instanci

##### Zapouzdření

- umožňuje skrýt některé metody a atributy (nazývají se pak vnitřní)
- provádí se podtržítkem, nebo dvěma
- s jedním podtržítkem jde jen o domluvenou konvenci

##### Magické metody objektů

- jsou předdefinované metody objektů a začínají a končí dvěma podtržítky
- `__new__` slouží pro lepší kontrolu nad vytvářeným objektem
- a vrací buď vytvořený objekt a volá `__init__`, nebo nic
- `__init__` volá se při inicializaci objektů
- `__del__` destruktorka – volá se při zničení objektu
- Reprezentační metody:
- `__repr__` str pro vytvoření objektu
- `__str__` str pro reprezentaci objektu
- `__bytes__` vrátí reprezentaci objektu za pomoci bitů
- `__hash__` při použití `__eq__` musí být předdefinovaná
- `__bool__` vrací True/False
- `__call__` umožňuje volat objekt, jakoby byla funkce
- Porovnávací magické metody:
- `__lt__`, `__le__`, `__eq__`, `__ne__`, `__gt__`, `__ge__` == (<, <=, ==, !=, >, >=)
- Matematické magické metody
- `__add__`, `__sub__`, `__mul__`, `__truediv__` == (+, -, \*, /)
- `__floordiv__`, `__mod__`, `__divmod__`, `__pow__` == (//, %, //+%, \*\*)
- `__lshift__`, `__rshift__`, `__and__`, `__or__`, `__xor__` == (<<, >>, &, |, ^)
- Magické funkce pro prohozené operátory
- Magické funkce pro zkrácenou notaci
- Magické funkce pro kolekce a deskriptory
- Magické metody pro řízení atributů

##### Dědičnost a polymorfismus

- Dědičnost je základní vlastnost OOP a vytváří nové datové struktury na základě starých
- Polymorfismus umožňuje používat jednotné rozhraní pro práci s různými typy objektů
- Polymorfismus funguje na základě překrývání metod

##### Abstraktní třída

- je třída u které nemá smysl vytvářet instanci a slouží pouze jako vzor pro třídu s instancí
- `from abc import ABC, abstractmethod / class NázevTřídy(ABC):`

- dekorátor `@abstractmethod` se používá k označení metod, které musí být překryty

##### Vlastnosti

- mění výstup metody na atribut
- používají se k tomu dekorátory `@...`
- `@property` slouží k pro vytvoření vlastností z getteru a setteru
- `__dict__` ( `__slots__` ) je slovník kde můžeme přepsat atribut, pokud ho chceme veřejný

##### Statika

- jsou třídní proměnné a metody nezávislé na instanci
- jedná se o pomocné metody které se volají na třídě
- `@staticmethod` vs `@classmethod`

## Programátor www aplikací v Pythonu

### Okruh 3 - SQLite a Kolekce

Každý účastník dostane 6 otázek.

#### SQLite

##### CRUD (INSERT, SELECT, UPDATE, DELETE)

- sqlite je relační databáze (založená na tabulkách)
- RDBMS = Relation database management systém
- ACID – Nedělitelnost, Konzistence, Izolace, Okamžité zapsání
- SQL – struktura Query language

##### Základní SQL funkce

- CREATE TABLE, DROP TABLE, ALTER TABLE,
- SELECT / FROM / WHERE / GROUP BY / HAVING / ORDER BY (DESC)
- LIKE % / \_
- Agregční funkce: AVG, COUNT, MAX, MIN, SUM
- Textové funkce: LENGTH, LOWER, UPPER, SUBSTRING, REPLACE
- Numerické funkce: MOD, ROUND, CEIL, FLOOR, TRUNCATE, ABS, SQRT

##### Datové typy

- INTEGER, REAL, Text, BLOB, NULL

##### JOIN

- slouží k dotazu přes více tabulek
- INNER JOIN – Vrací řádky které mají v v obou tabulkách odpovídající hodnoty, Nevrací NULL
- OUTER JOIN (LEFT, RIGHT, FULL, CROSS, NATURAL) – Vrací i ty výsledky, které se nepodařilo spojit
- namísto JOINU se dá dojít k stejnému výsledku i whereováním (WHERE...)

##### Databázové vazby (relace)

- 1:1 jeden uživatel má jednu věc
- 1:N k jednomu uživateli se váže více věcí
- N:M více uživatelů mohou navzájem vlastnit více věcí

##### Poddotaz

- je dotaz, který je součástí nějakého dalšího dotazu (SELECT v SELECTu)

##### Víceřádkové perátory

- IN umožňuje otestovat přítomnost hodnoty ve výčtu hodnot
- EXIST u korelovaných poddotazů umožňuje zjistit zda vrátil nějakou hodnotu

##### Transakce

- je soubor několika dotazů, které databáze chápe jako jeden dotaz
- začíná BEGIN TRANSACTION (nebo BEGIN)
- končí COMMIT, nebo END TRANSACTION

##### Pohledy

- slouží k ukládání často používaných nebo složitých SQL dotazů

##### Databázové indexy

- jsou indexy umožňující přímý přístup k datům
- může se jednat o primární klíče, nebo pomocné indexy
- FULLTEXTOVÉ VYHLEDÁVÁNÍ FUNGUJE PŘES INDEXY

#### Trigery

- je uložený program, který se spustí automaticky jako reakce na určitou akci s přidruženou tabulkou
- reagují na 3 druhy událostí: UPDATE, INSERT, DELETE

#### Kolekce v Pythonu

##### N-tice

- je seznam ve kterém nejde dále položky modifikovat (neměný seznam)
- inicializují se tuple(), nebo jen závorkami (), při jedné hodnotě s čárkou

##### Množiny

- seznam obsahující pouze unikátní položky
- metody: add(), remove(), discard(), pop(), clear()
- difference(), difference\_update(), intersection(), isdisjoint(), issubset() a issuperset()

##### Slovníky

- je množina skládající se z klíče a hodnoty
- metody: get(), values(), keys(), items(), clear()

##### Seznamy - Vícerozměrná pole

- Dvourozměrný seznam si můžeme v paměti představit jako tabulku
- N-rozměrné seznamy jsou seznamy s více než dvojitým vnořením (3D pole)
- Zubaté seznamy – částečná tabulka

##### Komprehence

- umožňuje stručný zápis kolekce z iterovatelného objektu za pomoci transformace, iterace a případně filtru
- syntaxe pro seznam: novy\_seznam = [prvek \* 2 for prvek in jiny\_seznam if prvek < 5]

##### Lambda výrazy

- jedná se o zkrácený zápis jednořádkové funkce
- začíná slovem lambda a výsledná funkce nemá jméno

##### ChainMap

- je datový typ, který nám umožňuje sdružovat více slovníků do jednoho logického celku

##### NamedTuple

- jedná se o speciální třídu neseřazeného datového typu, která kombinuje výhody tuple a slovníku.

##### DeQue

- třída představuje specifický typ fronty, který je implementovaný jako dvoustranně otevřený seznam.

##### Counter

- je speciální slovník, kde jednotlivé prvky jsou uloženy jako klíče a četnost těchto prvků jako příslušné hodnoty.

##### OrderedDict

- třída, kde prvky jsou uloženy v takovém pořadí, v jakém byly postupně do slovníku přidávány

##### defaultdict

- třída poskytuje slovník s výchozí hodnotou pro klíče, které ještě neexistují v daném slovníku

# Programátor www aplikací v Pythonu

## Okruh 4 - Webdesign

Každý účastník dostane 3 otázky.

### Moderní webdesign

#### Boxmodel

- krabičkový model
- margin / border / padding / element

#### Struktura HTML stránky

- <!doctype> / <html><head></head><body><body></body></html>

#### Blokové/řádkové elementy (Párové/nepárové elementy)

- párové elementy se zobrazují na celou šířku rodičovského elementu
- skládají se ze začátku a konce a mezi ně se vkládá obsah
- nepárové elementy se vkládají do párových

#### Tabulky

- ohraničení tabulky <table border="1"></table>
- hlavička tabulky <thead></thead>
- řádek a buňka hlavičky (th) - <tr><th>popis sloupců</th></tr>
- tělo tabulky <tbody></tbody>
- řádek a buňka těla (td) - <tr><td>Buňka</td></tr>
- patička tabulky <tfoot></tfoot>
- colspan/rowspan = slučování buňek

#### Seznamy

- <ul> = Unorder List - neuspořádaný seznam – položky jsou neočíslované
- <ol> = Ordered List - uspořádaný seznam - položky jsou řazeny dle klíče
- <li> = List Item - položka seznamu

#### Slovníček pojmů:

- <dl> = Definition List
- <dt> = Definition term – vysvětlovaný pojem
- <dd> = Definition Definition – vysvětlení pojmu

#### Flexbox

- částečně nahrazuje vytváření sloupečků pomocí float
- umožňuje měnit pořadí sloupců vedle sebe

#### Flexbox Grid

- slouží na rozložení obsahu stránky na webu
- breakpointy – xs, sm, md, lg

#### Frameworky

- jedná se o ucelené knihovny, nebo sady knihoven, které dávají dohromady celá řešení webových stránek

Framework Django není součástí zkoušky.