

BULUTLI TEXNOLOGIYALARNING PHYTON DASTURLASH TILIDA AFZALLIKLARI

Otaxonov Nurillo Abdumalikovich
Namangan Davlat Universiteti professori

Zarifa Turdaliyeva Dilshodbek qizi
Namangan Davlat Universiteti Amaliy matematika mutaxassisligi 1–kurs magistranti

ANNOTATSIYA

Yaratilganidan beri kompyuter tarmoqlari oldinga katta qadamlar qo‘ydi. Ikki kompyuter o‘rtasida ma’lumotlar bitlarini uzatishning eng oddiy operatsiyalaridan boshlab, butun dunyo bo‘ylab soatiga minglab terabaytgacha, simlar va simsiz tarmoqlar bilan qoplangan. Veb-texnologiyalarni takomillashtirish va dasturlash tillarini o‘zlashtirishning nisbatan qulayligi murakkab hisoblash muammolarini faqat Internetga kirish bilan hal qilish imkonini beradi.

Kalit so‘zlar: Bulutli texnologiyalar, SaaS, PaaS, IaaS, CaaS, shaxsiy bulut, umumiy bulut, gibril bulut

ПРЕИМУЩЕСТВА ОБЛАЧНЫХ ТЕХНОЛОГИЙ НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ PYTHON

Отаханов Нурилло Абдумаликович
Профессор Наманганского государственного университета

Зарифа Турдалиева Дилшодбек кизи
Наманганский государственный университет, магистрант 1 курса,
специальность “Прикладная математика”

АННОТАЦИЯ

С момента своего создания компьютерные сети шагнули далеко вперед. От простейших операций по передаче битов данных между двумя компьютерами до тысяч терабайт в час по всему миру охватывают провода и беспроводные сети. Совершенствование веб-технологий и относительная простота освоения языков программирования позволяют решать сложные вычислительные задачи, имея лишь доступ к Интернету.

Ключевые слова: Облачные технологии, SaaS, PaaS, IaaS, CaaS, частное облако, публичное облако, гибридное облако.

ADVANTAGES OF CLOUD TECHNOLOGIES IN PYTHON PROGRAMMING LANGUAGE

Otakhanov Nurillo Abdumalikovich

Professor of Namangan State University

Zarifa Turdaliyeva Dilshodbek kizi

Namangan State University, 1st-year master's student, majoring in applied
mathematics

ABSTRACT

Since its inception, computer networks have made great strides forward. From the simplest operations of transferring bits of data between two computers, to thousands of terabytes per hour around the world, wires and wireless networks cover it. The improvement of web technologies and the relative ease of mastering programming languages make it possible to solve complex computing problems with only access to the Internet.

Keywords: Cloud technologies, SaaS, PaaS, IaaS, CaaS, private cloud, public cloud, hybrid cloud

KIRISH

O'tgan asrning 60-yillarida muhokama qilina boshlagan bulutli texnologiyalar (inglizcha Cloud - bulutdan) so'nggi bir necha yil ichida mo'ljallangan shaklda mumkin bo'ldi va ishlab chiquvchilar tomonidan eng arzon va xavfsiz yechimlar sifatida joylashtirildi.

Bulutli hisoblash oxirgi foydalanuvchi kompyuteridan yukni olib tashlash va axborotni qayta ishlashni Internetga ulangan kuchli hisoblash klasteriga o'tkazish va ma'lumotlarni qayta ishlash uchun resurslarni taqdim etish imkonini beradi. Bunday holda, foydalanuvchi faqat dasturga kiritilgan va undan qaytgan ma'lumotlarga kirish huquqiga ega. Shunday qilib, ilgari faqat yirik korporatsiyalar uchun mavjud bo'lgan ma'lumotlar operatsiyalari endi global tarmoqqa kirish imkoniga ega bo'lgan har bir kishiga xizmat qiladi.

ASOSIY QISM

Ushbu ishda men bulutli texnologiyalarning paydo bo'lishi, ularning ishlash tamoyillari, Python tilidan foydalangan holda bulutli hisoblashni amalga oshirish usullari haqida gapirmoqchiman, shuningdek, ushbu til yordamida yaratilgan bulutli ma'lumotlarni qayta ishlash tizimini yaratish misolini ko'rib chiqmoqchiman.

1. Bulutli texnologiyalar-Hozirgi vaqtda bulutli tizimlar xizmatlariga murojaat qilmaydigan shaxsiy kompyuter foydalanuvchilari deyarli qolmadi. Tarmoqqa biriktirilgan ma'lumotlarni saqlashdan tortib, o'ta murakkab hisoblash uchun ko'p bosqichli kadrlargacha, ko'rinmas ko'rinadigan "bulutlar" zamonaviy hayotda mustahkam o'rnatilgan.

Bulutli hisoblash – bu texnologiya bo'lib, unda oxirgi foydalanuvchi apparat vositalaridan foydalanmasdan yoki ishchi dastur kodini o'zgartirish imkoniyatiga ega bo'lmasa ham, qayta ishlangan ma'lumotlar ustidan to'liq nazoratga ega.

Bu atama 1960-yillarda Jon Makkarti tufayli paydo bo'lgan. Bu olim o'z vaqtida "sun'iy intellect" atamasidan foydalanishni taklif qilgan, Lisp dasturlash tilini ixtiro qilgan va funksional dasturlash asoslarini yaratgan va Tyuring mukofoti laureati bo'lgan. Yarim asrdan ko'proq vaqt oldin, MakKarti "foydali uslub" tizimining hisoblash resurslaridan foydalanish bo'yicha loyiha prototipini yaratdi - biznes xizmatlari modelida ishlaydigan vaqtning taqsimlovchi kompyuter texnologiyasi, hozirgi suv va elektr energiyasini taqsimlash usuli. Bu g'oya 1970-yillargacha mashhur bo'lgan, biroq o'sha davr texnologiyasi bunga yo'l qo'ymasligi ma'lum bo'lgach, loyiha deyarli 40 yil davomida unutilgan edi.

2008 yilda birinchi bulutli texnologiyalarni yaratish bo'yicha muvaffaqiyatli urinishlar boshlandi. Taxminlarga ko'ra, bulut atamasi birinchi marta qidiruv tizimi sifatida boshlangan, ammo hozirda veb-texnologiyaning ko'plab sohalarida yo'nalishlarni belgilab beruvchi media giganti Google rahbari Erik Shmidt tomonidan kiritilgan. Boshqa versiyaga ko'ra, bu atama 2005 yilda eng yirik auktsion tizimlaridan biri Amazon.com Elastic Compute Cloud (Amazon EC2) xizmatini ishga tushirgandan so'ng mashhur bo'lgan. EC2 veb-serverlarni yaratish va boshqarish uchun eng oson sozlash va eng qulay bulut xizmatlaridan biri bo'lib, bugungi kunda ham ishlaydi.

"Bulut" shuningdek, metaforik ravishda butun Internetning sxematik tasvirini yoki barcha ahamiyatsiz tafsilotlar yashiringan murakkab tuzilmalar majmuasini anglatadi.

2. Bulutli texnologiyalarning xususiyatlari - Zamonaviy bulutli texnologiyalar barcha turdagi vazifalarni bajarish uchun komplekslardir. Asosiy muammoni hal qilishda nazariy jihatdan umumiy yondashuvga qaramasdan (ishni tezlashtirish uchun vaqt va resurslarni taqsimlash), texnik jihatdan tizimlar ko'plab omillar bilan farqlanadi. AQSh Milliy standartlar va texnologiyalar instituti bulutli hisoblashlarga xos bo'lgan asosiy xususiyatlarni aniqladi:

- Talab bo'yicha o'z-o'ziga xizmat ko'rsatish - foydalanuvchining o'zi server vaqti, tezligi va qayta ishlangan ma'lumotlarga kirish, ma'lumotlarni saqlash uchun zarur bo'lgan ma'lumotlar miqdori kabi o'zgaruvchilarni aniqlaydi va o'zgartiradi.

- Tarmoq orqali universal kirish - "bulutlarga" tarmoqqa kirish imkoni bo'lgan har qanday qurilmadan kirish mumkin.

- Resurslarni birlashtirish - bulutli xizmat ko'rsatuvchi provayder tizim resurslarini hovuz deb atalmish - dinamik yuk taqsimoti uchun klasterga birlashtiradi. Keyin foydalanuvchilar talab bo'yicha o'z-o'ziga xizmat ko'rsatish tamoyiliga muvofiq, kerakli quvvatni, tizim tezligini va kerakli xotira hajmini aniqlaydilar.

- Elastiklik – foydalanuvchi istalgan vaqtda hech qanday kechikishlarsiz xizmatlar ko'rsatish va o'zgartirish imkoniyati.

- Iste'molni hisobga olish - etkazib beruvchi vazifani bajarish uchun zarur bo'lgan resurslar miqdorini avtomatik hisoblashni ta'minlaydi. Ushbu ma'lumotlarga asoslanib, masalan, agar xizmat to'langan bo'lsa, xizmatlardan foydalanishning yakuniy narxi e'lon qilinadi.

Zamonaviy IT-fanning eng ilg'or yo'nalishi bo'lgan bulutli texnologiyalar doimiy ravishda rivojlanayotgan kuchli xizmatlar, ya'ni aaS (xizmat sifatida) tomonidan xizmat qiladi:

- SaaS (xizmat sifatida dasturiy ta'minot) – obuna orqali foydalanuvchiga ma'lum dasturiy ta'minot to'plamiga kirish huquqi beriladigan xizmat. Misol tariqasida, Google va uning Google Docs.

- PaaS (xizmat sifatida platforma) – maqsadli auditoriyasi oxirgi foydalanuvchi emas, balki ishlab chiquvchi bo'lgan xizmat. PaaS dasturchiga vazifani hal qilish uchun zarur bo'lgan kerakli komponentlar va kutubxonalar bilan birga butun platformani taqdim etadi. Misol: Google AppEngine, agar kerak bo'lsa, qo'shimcha hisoblash resurslarini ham sotib olishingiz mumkin.

- IaaS (xizmat sifatida infratuzilma; xizmat sifatida infratuzilma) – virtualizatsiya va emulyatsiya texnologiyalaridan foydalangan holda foydalanuvchini bir butunga birlashtirilgan apparat resurslari bilan ta'minlaydigan xizmat (hisoblash klasteri – foydalanuvchi tomonidan ma'lum komponentlar va dasturiy ta'minot sifatida ishlash uchun tuzilgan dasturiy ta'minot. belgilaydi). Misollar: yuqorida aytib o'tilgan Amazon EC2, shuningdek, tarmoq ma'lumotlarining barcha turlari: Dropbox va boshqalar.)

- CaaS (xizmat sifatida aloqa) eng mashhur xizmatdir. CaaS elektron pochta, ijtimoiy tarmoqlar va tezkor xabar almashish dasturlarida qo'llaniladi.

3. Bulutli texnologiyalarni joylashtirish modellari - "bulutlar" umumiy printsiplarga muvofiq ishlaydi, ammo turli funktsiyalarni bajarishda ular har bir vazifani hal qilishda individual yondashuvni hisobga olgan holda tashkil etilishi kerak. Bulutni joylashtirishning quyidagi asosiy modellari ajralib turadi:

- Shaxsiy bulut - korporativ vazifalarni bajarish uchun bir tashkilot ichida va uchun yaratilgan tuzilma. Ko'pincha elektron hujjatlarning ishlashini va / yoki korxon

ustidan nazoratni ta'minlash uchun ishlatiladi. Korporativ egasi, texnologiya provayderi yoki uchinchi tomon tomonidan nazorat qilinadi.

- Umumiy bulut – umumiy foydalanish uchun mo'ljallangan tuzilma. Tijorat, ilmiy yoki davlat tashkilotlari tomonidan boshqarilishi mumkin. Mavjud va etkazib beruvchiga bog'liq.

- Gibril bulut – ikki yoki undan ortiq bulutli tuzilmalarning sintezidan iborat bo'lib, odatda bir nechta tuzilmalarning funksiyalari zarur bo'lganda oraliq variant hisoblanadi. U asosiy tarmoqlarni tushirish sharoitida ham yaratilishi mumkin.

- Ommaviy bulut - bu o'xshash xizmatlarga (masalan, xavfsizlik) umumiy talablarga ega bo'lgan iste'molchilar yoki tashkilotlarning ma'lum bir jamiyati tomonidan foydalanishga mo'ljallangan tuzilma. Bunday bulutlar ommaviy ravishda boshqarilishi mumkin.

4. Bulutli texnologiyalarning texnologik jarayonlari, iqtisodiy jihatlari va tanqidi -Yetkazib beruvchi ham, iste'molchi ham bulutli texnologiyalarni joriy etishdan iqtisodiy foyda ko'radi: iste'molchi hisoblash ma'lumotlar markazlari uchun qimmat ijara haqi to'lamaydi, yetkazib beruvchi esa ularni bir xilda qimmat qurilish va texnik xizmat ko'rsatish bilan shug'ullanishi shart emas, chunki bunday murakkab infratuzilmalar bir vaqtning o'zida bir vaqtning o'zida bir vaqtning o'zida bir vaqtning o'zida bir vaqtning o'zida ishlab chiqarish quvvatini iste'mol qilishi kerak. katta hajmdagi tarmoq resurslari.

Bulutli sxemadan foydalanganda ma'lumotlarni uzatishning deyarli butun yoki Internetga ulanish provayderiga tushadi. Bulutli xizmatlardan foydalanishning iqtisodiy komponentini tavsiflash uchun ko'pincha elektr, suv, gaz ta'minoti xizmatlariga o'xshashlik keltiriladi: iste'molchi faqat iste'mol uchun, masalan, suv uchun to'laydi. U nasos sotib olishi va o'rnatishi yoki uyiga barcha quvurlarni o'rnatishi shart emas.

Bitta tarmoqqa ulangan kompyuterlar o'zlarining hisoblash resurslarini mustaqil ravishda baham ko'rmaydilar, xuddi ko'p yadroli protsessorlar ushbu qurilmalarni dasturlashda to'g'ri yondashmasdan ishlashning o'sishini ta'minlamaydi.

Bulutli xizmatlar, shuningdek, apparat va dasturlarning monitoringini ta'minlaydigan maxsus vositachi dastursiz mavjud bo'lolmaydi. Bunday dasturiy ta'minot yukni bulutli ramka quyi tizimlari o'rtasida taqsimlaydi va "tezda" vazifalarni bajarish uchun zarur bo'lgan bulut tomonidan taqdim etilgan resurslarni taqsimlaydi. Bu bizga bulutlar sezgir bo'lgan asosiy muammoni - notekis yukni engish imkonini beradi. Murakkab ko'p bosqichli bulut xizmati dasturiy ta'minoti murakkab vazifalarni avtomatik ravishda klasterning bo'sh va yuklanmagan joylariga o'tkazadi, shu bilan birga boshqa hududlardan resurslarni olib tashlamaydi.

Oddiy soʻzlar bilan aytganda, bitta foydalanuvchi oqsil molekulasini hisoblash vazifasini bulutga oʻtkazganda, sizning Pi raqamingizni hisoblash sekinlashmaydi, siz bulut oʻz pozitsiyasi va tuzilishini deyarli oʻzgartirganini sezmaydiz.

Bulutlar kontseptsiyasi va amalga oshirilishi, uning barcha afzalliklariga qaramay, tanqiddan chetda qolmadi. Erkin dasturiy taʼminot harakati asoschisi Richard Stallman foydalanuvchi “bulutli veb-ilovani boshqarishni yoʻqotib qoʻyishi” haqida bir necha bor eslatib oʻtgan.

Bulutli xizmatlardan foydalanganda, foydalanuvchi birinchi navbatda yakuniy natija bilan qiziqsa ham, maʼlumotlar nazoratisiz bulutlarning nomaʼlum sektorlarida cheksiz saqlanadigan vaziyat yuzaga keladi, deb taxmin qilish mumkin emas. keraksiz yuk yoki foydalanuvchining maʼlumotlari notoʻgʻri qoʻllarga tushib qolsa, xavfsizligiga tahdid solishi. 2010-yildayoq, Strategic News Service xodimi Mark Anderson kabi hurmatli tanqidchilar maʼlumotlarning sizib chiqishi va maʼlumotlarni qayta ishlash xatolarining narxi oshishini bashorat qilishgan.

Biroq, 2013 yil holatiga koʻra, maʼlumotlarning tarqalishi yoki maʼlumotlar yoʻqolishining koʻplab asosiy holatlari maʼlum emas (2009 yilda Magnolia xatchoʻplarni saqlash xizmati barcha maʼlumotlarning yoʻqolishi haqida xabar bergan). Buni ushbu xizmatlarning mashhurligining oʻsishiga oid notoʻgʻri prognozlar, shuningdek, yukni muvaffaqiyatli engadigan hisoblash klasterlarining etarli kuchi bilan izohlash mumkin. Shu bilan birga, bulutlardan maʼlumotlarni oʻchirishning mumkin emasligi haqidagi xavotirlarni vaqtinchalik deb hisoblash mumkin, chunki bu texnologiya hali juda yosh va vaqt oʻtishi bilan unga koʻp narsa qoʻshiladi. Faqatgina ushbu xizmatlarni etkazib beruvchilar oʻrtasidagi raqobat kuchayganligi sababli.

5. Python dasturlash tili - Python yordamida amalga oshirilgan bulut tizimini tahlil qilishdan oldin, keling, tilning oʻzini koʻrib chiqaylik.

Python (Python, Britaniyaning Monty Python komikslar guruhiga havola) golland dasturchisi Guido van Rossum tomonidan ishlab chiqilgan va yaratilgan. Python dasturchilari hazil bilan “loyihaning umrbod xayrixoh diktatori” deb atashadi. Bu shuni anglatadiki, Guido tildagi barcha oʻzgarishlarni kuzatib boradi va vaziyat talab qilganda muayyan funktsiyalarni amalga oshirish boʻyicha yakuniy qarorni qabul qiladi.

Van Rossum shuningdek, ABC taʼlim dasturlash tilini ishlab chiqishda ishtirok etgan, 2001 yilda nufuzli Free dasturiy taʼminot mukofotini qoʻlga kiritgan va Googleʼda ishlagan. Endi Python yaratuvchisi Dropbox inc da ishlaydi, yaʼni u bulutli xizmatlar bilan bevosita bogʻlangan.

Til 1991 yilda paydo boʻlib, oʻzini koʻp paradigma deb eʼlon qildi (dasturlar bir xil tilda, lekin turli uslublarda yozilgan). Obʼektga yoʻnaltirilgan (maydonlar va usullar bilan ishlaydi). Kross-platforma (Python-da siz Windows va MacOS, linux, *nix va

boshqa mashhur operatsion tizimlarda bir xil imkoniyatlar to'plami bilan dasturlashingiz mumkin).

Reflektiv (dastur o'z tuzilishini tahlil qilishi va kod bajarilishi bilan uni o'zgartirishi mumkin). Imperativ (to'g'ridan-to'g'ri ko'rsatmalar, "buyruqlar" dan keyin). Funktsional (ma'lumotlarni ramziy qayta ishlash). Aspektga yo'naltirilgan (dastur modul-aspektlarga bo'lingan).

Ya'ni, minimalist sintaksisga ega bo'lgan holda, u kattaroq dasturlashtiriladigan muhitlardan kam emas va ba'zan undan ham oshib ketadi. Ushbu minimalizm sizga dasturlarni yozish tezligini oshirish, shuningdek, kodni o'qish imkonini beradi. Shu bilan birga, modullarning standart kutubxonasi doimiy ravishda o'sib borayotgan turli xil funktsiyalar to'plamini o'z ichiga oladi va foydalanuvchi etishmayotganlarini osongina qurishi mumkin.

6. Til falsafasi - Boshqa dasturlash tillarida bo'lgani kabi, dasturlar sintaksis deb ataladigan maxsus qoidalar yordamida yozilishi kerak. Shu bilan birga, Python dasturchilari ham Python Zen deb ataladigan deyarli dogmatik ko'rsatmalarga amal qilishadi:

- Xunukdan go'zal yaxshi
- Yopiqdan ko'ra, ochiq-oydin.
- Murakkabdan oddiy narsa afzal
- Murakkablik chalkashlikdan yaxshiroqdir
- Yassi ichki o'rnatilgandan yaxshiroqdir
- Zichlikdan ko'ra siyraklik afzalroqdir
- O'qilishi muhim.
- Qoidalarni buzadigan darajada maxsus bo'lmagan alohida holatlar (istisnolar).
- Amaliylik mukammallikdan muhimroqdir
- Xatolarni yashirmaslik kerak
- Agar ular aniq jim bo'lmasa
- Noaniqlik bilan duch kelganda, taxmin qilish vasvasasidan voz kechishga

arziydi

- Buni amalga oshirishning bitta aniq yo'li bo'lishi kerak
- Gollandiyalik emasligingiz aniq bo'lmasligi mumkin
- Hozir hech qachondan yaxshiroq
- Garchi hech qachon hozirgidan yaxshiroq bo'lmasa ham
- Agar amalga oshirishni tushuntirish qiyin bo'lsa, g'oya yomon
- Agar bu oson bo'lsa, ehtimol yaxshi
- Ismlar maydoni - bu ajoyib narsa! Keling, ulardan ko'proq narsani yarataylik!

Ushbu yarim hazil qoidalaridan quyidagi xulosa kelib chiqadi: Python dasturi, birinchi navbatda, amalga oshirish oson va tuzilishi oson bo'lishi kerak, lekin agar vaziyat buni talab qilsa, dasturchi kodni qanday tuzishni o'zi hal qilishi mumkin.

PEP8 - Qoidalar to'plamiga PEP8 deb ataladigan ham kiradi. Bu Python kodini yozish uchun umumiy uslub qoidalari to'plami. PEP8 batafsil kodlash ko'rsatmalari, umumiy maslahatlar va amalga oshirish misollari bilan tez-tez so'raladigan savollarga ega ko'plab sahifalardan iborat.

Garchi bu yangi dasturchining eng asosiy muammolarini hal qilishga yordam berishi mumkin bo'lsa-da, ko'p hollarda dasturchilar jamoalari o'zlarining PEP8 ga qo'shib, har bir a'zo va umuman jamoaning mahsuldorligini oshiradilar. Quyida e'tiborga olinishi kerak bo'lgan ba'zi qoidalarni ko'rib chiqamiz:

- Chiziq uchun 4 ta bo'sh joydan foydalaning, yorliqlardan foydalanmang, yorliqlarni (Tab tugmasi) va bo'shliqlarni aralashtirmang.

- Maksimal qator uzunligi 79 belgidan iborat. Buzish uchun chekinish va teskari chiziqdan foydalaning (`\` tugma).

- Yuqori darajadagi funktsiyalar va sinf ta'riflari ikkita bo'sh qator bilan ajratilishi kerak.

- Import qilingan har bir modul yangi qatordan boshlanadi

- Qavslar ichida vergul, nuqtali vergul va ikki nuqtadan oldin qo'shimcha bo'sh joy qoldirmang.

- Sharhlarni kod bilan yangilang

- Ularni ingliz tilida yozing

- I, O bir harfli ismlardan saqlaning

- Batafsil o'zgaruvchilar nomlarini bering

- Mantiqiy o'zgaruvchilarni True va False bilan solishtirmang

Hatto eng oddiy qoidalarga rioya qilgan holda ham, Python dasturchisi o'qish va tahlil qilish uchun teng darajada oson kod yaratadi. Yozishga bunday yondashuv Pythonning barcha dasturlash tillari orasida asosiy afzalliklaridan biridir.

7. Python tilining xususiyatlari- Kodning aniq, ammo juda oddiy tuzilishi tilning mashhurligi sababi. Yangi sinflar, usullar yoki halqalarni bildiruvchi noqulay konstruktsiyalarning yo'qligi bo'shliqlar va yorliqlar bilan vizual ta'kidlash bilan almashtiriladi. Bunday sharoitda dasturni yaratish jarayonini kuzatish ancha oson, uni disk raskadrovka qilish va kengaytirish osonroq. Strukturaviy Python kodini dasturlashda yangi boshlanuvchilar uchun ham, dasturlash tilini o'zgartirgan mutaxassis uchun ham tushunish ancha oson. Shu bilan birga, shuni ta'kidlash kerakki, hech bo'lmaganda ob'ektga yo'naltirilgan dasturlash bo'yicha nazariy bilimlarsiz Python va boshqa tillarni bilish qiyin bo'ladi.

Ammo turli dasturlash tillarini ishlab chiquvchilar o'rtasidagi shiddatli raqobat sharoitida muvaffaqiyatga erishish uchun chiroyli dizayn etarli emas. Quyidagi omillar ham Pythonning jahon sahnasidagi muvaffaqiyatining kalitidir:

1. Tilning talqin qilinishi. Xuddi Lisp va Prolog tillari bilan ishlaganda, foydalanuvchi uchun turli xil tarjimonlar to'plami mavjud. Bular dasturlash tilining grafik interfeyslari bo'lib, til bilan ishlashni osonlashtiradi. Misol uchun, standart Python tarqatish allaqachon "issiq" rejimda dasturchi tomonidan kiritilgan har bir buyruqni bajaradigan tarjimonga ega.

2. Ob'ektga yo'naltirilgan yondashuv. OOP Python amalga oshiriladigan asosiy modeldir, lekin bu model an'anaviydan biroz farq qiladi: Sinflar dastur ichidagi ob'ektlar bo'lishi mumkin va bo'lishi mumkin; Bir nechta meros qo'llab-quvvatlanadi; Virtual sinf polimorfizmi; Barcha darajadagi inkapsulyatsiya; Asosiy yig'ilishda konstruktorlar, destruktorlar va axlat yig'uvchilarning mavjudligi; Deyarli barcha operatorlarni, hatto tizimli operatorlarni bepul ortiqcha yuklash; Rivojlangan emulyatsiya qobiliyatlari; Eng tez-tez bajariladigan operatsiyalar bilan ishlashning "o'rnatilgan" usullari; Metadasturlash; Statik usullar, maydonlar va sinflar;

3. Dasturlash funksiyasi. Ob'ekt sifatida funktsiya bilan ishlash, to'liq rekursivlik, yopilishlar, funktsiyalar qismlariga kirish va o'z vositalarini yaratish imkoniyati mavjud.

4. Til kengaytmalarining paket-modul tizimi. Python, o'zining soddaligi tufayli, kerak bo'lganda tilning imkoniyatlarini moslashuvchan ravishda oshiradigan ko'plab qo'shimchalar va kengaytmalarga ega. Yangi modullarni ulash kod yoki tarjimonda bitta buyruqda amalga oshiriladi.

5. Introspeksiya. Ushbu til xususiyati so'rov bo'yicha, hatto ishlaydigan dastur ichidagi har qanday ob'ekt haqida batafsil ma'lumot olish imkonini beradi.

Bu va boshqa ko'plab funktsiyalar allaqachon tilning bir qismidir va ko'plab "og'ir" muhitlar qo'shimcha modullarni talab qiladigan joylarda Python o'zining ancha ixcham bazaviy taqsimoti bilan shug'ullanadi. Agar kerakli qo'shimcha mavjud bo'lmasa, uni o'rnatish yoki o'zingiz yozish ko'p vaqt talab qilmaydi.

XULOSA

Ishda belgilangan maqsad va vazifalarga erishildi. Bulutli texnologiyalarning umumiy ko'rinishi, ularning paydo bo'lish tarixi, ularni joylashtirish va amalga oshirish tamoyillari keltirilgan. Python tilining tavsifi ham ob'ektga yo'naltirilgan dasturlash nuqtai nazaridan, ham ushbu tilning texnik tomondan afzalliklari. Bulutli hisoblash sohasidagi eng mashhur yechimlarning tavsifi va mashhur Apache majmuasidan foydalangan holda bulutli xizmatlardan biri bilan ishlashni amalga oshirishning umumiy misoli berildi.

Qisqacha xulosalar:

Bulutli ma'lumotlarni qayta ishlash - bu yarim asr oldin, birinchi kompyuter tarmoqlarining prototiplari paydo bo'lganida, olimlarning g'oyasi yonib ketgan yosh texnologiya. Texnik baza yaqinda paydo bo'ldi, ammo bu bulutli hisoblashning zamonaviy IT fanidagi eng muhim sohalardan biriga aylanishiga to'sqinlik qilmadi.

Ushbu sohadagi barcha echimlar qat'iy asosiy qoidalar to'plamiga bo'ysunadi va ma'lum naqshlarga muvofiq joylashtiriladi. "Bulutlar" ixtisoslashtirilgan strukturaviy modellarga bo'lingan, ammo har bir loyiha uchun umumiy xususiyatlarga ega: foydalanuvchilar uchun keng qamrovlilik, foydalanish imkoniyati va keng boshqaruv imkoniyatlari.

Muvaffaqiyatli uzluksiz ishlashi uchun bulutli xizmatlar kuchli apparat va maxsus yozilgan oraliq dasturlarni talab qiladi, ularning vazifalari majburiy ravishda quvvatni dinamik taqsimlashni va uskunani oxirgi foydalanuvchi ehtiyojlariga moslashtirishni o'z ichiga oladi.

Python tili OOP paradigmalariga to'liq mos keladigan va modulli tuzilishi tufayli funktsionallikni kengaytirish uchun deyarli cheksiz imkoniyatlarga ega bo'lgan eng mashhur va tezkor skript dasturlash tillaridan biridir.

FOYDALANILGAN ADABIYOTLAR RO'YXATI: (REFERENCES)

1. Мурыгин К.В. Концепция системы автоматического распознавания номерных знаков автомобилей //Искусственный интеллект. – Донецк, 2012. – № 4 (58). – С. 220-226.
2. Трапезников И. Н. Разработка и анализ системы распознавания автомобильных регистрационных знаков: Автореф. дисс. – Минск: ЯГУ им. П.Г. Демидова, 2002. – 20 с.
3. Маллабоев Н., Абдуллаева Н. МЕСТО СИСТЕМЫ “ЭЛЕКТРОННОГО ПРАВИТЕЛЬСТВА” В РАЗВИТИИ МАЛОГО БИЗНЕСА И ПРЕДПРИНИМАТЕЛЬСТВА //Теория и практика современной науки. – 2016. – №. 6-1. – С. 834-838.
4. Abdullaeva N., Mamurova F., Mallaboev N. EFFICIENCY OF EXPERIMENTAL PREPARATION USE MULTIMEDIA TO ENLARGE SOME QUESTIONS //Экономика и социум. – 2020. – №. 6. – С. 11-13.
5. ПЕРСПЕКТИВЫ ВНЕДРЕНИЯ НОВЫХ ТЕХНОЛОГИЙ ДЛЯ РАЗВИТИЯ ТРАНСПОРТНОЙ СИСТЕМЫ РЕСПУБЛИКИ УЗБЕКИСТАН, ОК Касимов, СС Ражапова - Экономика и социум, 2020. – №. 6. – С. 710-715.
6. Абдулина Э.М. Облачные технологии в образовании // Молодой ученый. – 2019. – № 52 (290). – С. 7-9. – <https://moluch.ru/archive/290/65873>
7. Понятие «Облачные технологии» – https://studwood.ru/1046027/informatika/ponyatie_oblachnye_tehnologi
8. Облачные технологии: что это и как использовать бизнесу – <https://blog.sibirix.ru/tech-clouds>