

Ilovada oching ↗



Qidirmoq

Yozing



✦ Haftasiga \$1 dan kamroq evaziga eng yaxshi Mediumga cheksiz kirish huquqiga ega bo'ling. [A'zo bo'ling](#)

Eng yaxshi 50 ta muhim OOP intervyu savollari va javoblari

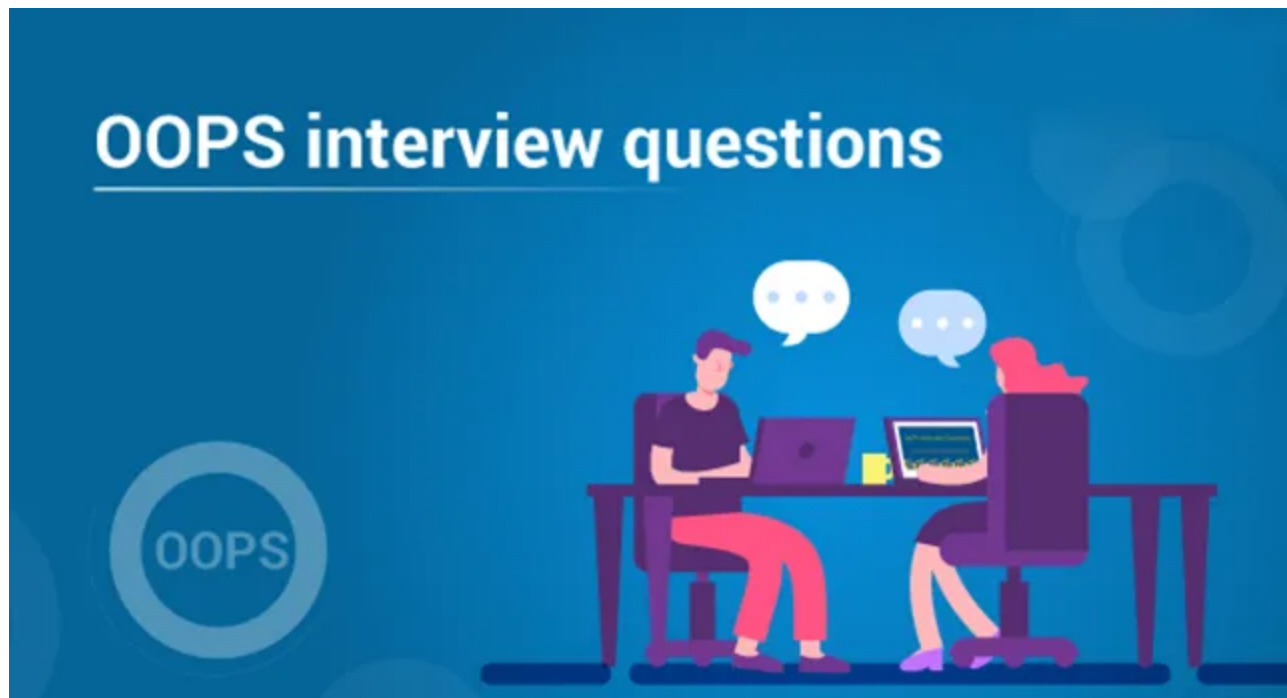


Aayushi Johari · Kuzatish

nashr etilgan Edureka · 10 daqiqa o'qish · 2019 yil 30 avgust



169



OOPS intervyu savollari - Edureka

OOPlar dasturlash dunyosidagi asosiy tushunchalardan biridir, shuning uchun siz qatnashgan har bir intervyu OOP haqida ma'lumotni talab qiladi. Ushbu maqola yangi talabalar uchun eng tez-tez so'raladigan OOP intervyu savollarini jamlaydi. Shunday qilib, oldinga boring va tayyorlaning.

Keling, muhokama qilinadigan barcha mavzularni tezda ko'rib chiqaylik:

- Asosiy OOPs intervyu savollari
- OOPs Intervyu savollari Sinflar va ob'ektlar
- OOP xususiyatlari bo'yicha intervyu savollari

1. Meros olish

2. Polimorfizm

3. Inkapsulyatsiya

4. Ma'lumotlarni abstraktsiya qilish

- Usul va funksiyalar Suhbat uchun savollar
- Intervyu savollariga istisnolardan foydalanish
- OOP cheklovlari

Asosiy OOP intervyu savollari:

1. OOP va SOP o'rtasidagi farq nima?

Object-Oriented Programming	Structural Programming
Object-Oriented Programming is a type of programming which is based on objects rather than just functions and procedures	Provides logical structure to a program where programs are divided functions
Bottom-up approach	Top-down approach
Provides data hiding	Does not provide data hiding
Can solve problems of any complexity	Can solve moderate problems
Code can be reused thereby reducing redundancy	Does not support code reusability

2. OOP nima?

OOP (ob'ektga yo'naltirilgan dasturlash) - bu shunchaki funktsiyalar va protseduralarga emas, balki ob'ektlarga asoslangan dasturlash turi. Alohida ob'ektlar sinflarga guruhlangan. OOPlar vorislik, polimorfizm, yashirish va boshqalar kabi real dunyo ob'ektlarini dasturlashda amalga oshiradi. Shuningdek, u ma'lumotlar va kodlarni birlashtirishga imkon beradi.

3. Nima uchun OOPlardan foydalanish kerak?

- OOPlar dasturlashda aniqlik va murakkab muammolarni hal qilishda soddalikka imkon beradi
- Kodni meros orqali qayta ishlatish mumkin, bu esa ortiqchalikni kamaytiradi
- Ma'lumotlar va kod bir-biriga inkapsulyatsiya orqali bog'langan
- OOP ma'lumotlarni yashirishga imkon beradi, shuning uchun shaxsiy ma'lumotlar maxfiy saqlanadi
- Muammolarni turli qismlarga bo'lish mumkin, bu ularni hal qilishni osonlashtiradi
- Polimorfizm tushunchasi ob'ektlarning bir nechta shakllarga ega bo'lishiga imkon berish orqali dasturga moslashuvchanlikni beradi

4. OOP ning asosiy xususiyatlari nimalardan iborat?

- Meros olish
- Inkapsulyatsiya
- Polimorfizm
- Ma'lumotlarni abstraktsiya qilish

Sinflar va ob'ektlar OOPs Intervyu savollari:

5. Ob'ekt nima?

Ob'ekt haqiqiy dunyo ob'ekti bo'lib, u OOPning asosiy birligi bo'lib, masalan, stul, mushuk, it va boshqalar. Turli xil ob'ektlar turli holatlar yoki atributlar va xatti-harakatlarga ega.

6. Sinf nima?

Sinf - bu turli holatlardagi va har xil xatti-harakatlarga ega ob'ektlardan iborat prototip. Bu sinfda mavjud bo'lgan ob'ektlar uchun umumiy bo'lgan bir qator usullarga ega.

7. Sinf va tuzilmaning farqi nimada?

Sinf: ob'ektlar yaratilgan foydalanuvchi tomonidan belgilangan reja. U ob'ektlarda bajarilishi kerak bo'lgan usullar yoki ko'rsatmalar to'plamidan iborat.

Struktura: Struktura asosan foydalanuvchi tomonidan belgilangan turli xil ma'lumotlar turidagi o'zgaruvchilar to'plamidir.

8. Misol yaratmasdan asosiy sinf usulini chaqira olasizmi?

Ha, siz asosiy sinfga uni yaratmasdan qo'ng'iroq qilishingiz mumkin, agar:

- Bu statik usul

- Asosiy sinf boshqa kichik sinf tomonidan meros qilib olingan

9. Sinf va ob'ekt o'rtasidagi farq nima?

Object	Class
A real-world entity which is an instance of a class	A class is basically a template or a blueprint within which objects can be created
An object acts like a variable of the class	Binds methods and data together into a single unit
An object is a physical entity	A class is a logical entity
Objects take memory space when they are created	A class does not take memory space when created
Objects can be declared as and when required	Classes are declared just once

OOP xususiyatlari bo'yicha intervyu savollari

Meros olish:

10. Meros degani nima?

Meros - bu OOPning xususiyati bo'lib, u sinflarga boshqa sinflardan umumiy xususiyatlarni meros qilib olish imkonini beradi. Misol uchun, agar "avtomobil" kabi sinf mavjud bo'lsa, "avtomobil", "velosiped" kabi boshqa sinflar avtomobil sinfidan umumiy xususiyatlarni meros qilib olishi mumkin. Bu xususiyat ortiqcha koddan xalos bo'lishga yordam beradi va shu bilan kodning umumiy hajmini kamaytiradi.

11. Merosning qanday turlari bor?

- Yagona meros
- Bir nechta meros
- Ko'p darajali meros
- Ierarxik meros

- Gibril meros

12. Ko'p va ko'p darajali merosning farqi nimada?

Multiple Inheritance	Multilevel Inheritance
Multiple inheritance comes into picture when a class inherits more than one base class	Multilevel inheritance means a class inherits from another class which itself is a subclass of some other base class
Example: A class defining a child inherits from two base classes Mother and Father	Example: A class describing a sports car will inherit from a base class Car which in turn inherits another class Vehicle

13. Gibril meros nima?

Gibril meros - bu ko'p va ko'p darajali merosning kombinatsiyasi.

14. Ierarxik irsiyat nima?

Ierarxik meros deganda bitta asosiy sinf bir nechta kichik sinflarga ega bo'lgan merosni anglatadi. Masalan, avtomobil sinfi o'zining kichik sinflari sifatida "avtomobil", "velosiped" va hokazolarga ega bo'lishi mumkin.

15. Merosning qanday cheklovlari bor?

- Dasturni bajarish uchun vaqt va kuchni oshiradi, chunki u turli sinflar orasida oldinga va orqaga sakrashni talab qiladi.
- Ota-onalar sinfi va bolalar sinfi chambarchas bog'lanadi
- Dasturga kiritilgan har qanday o'zgartirishlar ota-onalarda ham, bolalar sinfida ham o'zgarishlarni talab qiladi
- Ehtiyotkorlik bilan amalga oshirish kerak, aks holda noto'g'ri natijalarga olib keladi

16. Superklass nima?

Superklass yoki asosiy sinf boshqa sinf yoki sinflar uchun ota-ona vazifasini bajaradigan sinfdir. Masalan, Vehicle klassi Car sinfining yuqori sinfidir.

17. Kichik sinf nima?

Boshqa sinfdan meros bo'lib qolgan sinf pastki sinf deb ataladi. Masalan, Car sinfi pastki sinf yoki Avtomobil sinfidan olingan.

Polimorfizm

18. Polimorfizm nima?

Polimorfizm bir nechta shakllarda mavjud bo'lish qobiliyatini anglatadi. Bitta interfeysga bir nechta ta'riflar berilishi mumkin. Misol uchun, agar sizda Vehicle nomli sinfingiz bo'lsa, u tezlik deb nomlangan usulga ega bo'lishi mumkin, lekin siz uni aniqlay olmaysiz, chunki turli transport vositalarining tezligi har xil. Ushbu usul turli xil transport vositalari uchun turli xil ta'riflar bilan kichik sinflarda aniqlanadi.

19. Statik polimorfizm nima?

Statik polimorfizm (statik bog'lanish) kompilyatsiya vaqtida yuzaga keladigan polimorfizmning bir turi. Kompilyatsiya vaqti polimorfizmiga misol qilib, usullarni haddan tashqari yuklash mumkin.

20. Dinamik polimorfizm nima?

Ish vaqti polimorfizmi yoki dinamik polimorfizm (dinamik bog'lanish) polimorfizmning bir turi bo'lib, u ish vaqti davomida hal qilinadi. Ish vaqti polimorfizmiga misol usulni bekor qilishdir.

21. Usulni ortiqcha yuklash nima?

Usulni haddan tashqari yuklash - bu OOPning xususiyati bo'lib, agar o'tkazilgan argumentlar boshqacha bo'lsa, sinf ichidagi bir nechta usullarga bir xil nom berish imkonini beradi.

22. Usulning ustunligi nima?

Usulni bekor qilish - bu OOPning xususiyati bo'lib, uning yordamida bolalar sinfi yoki pastki sinf asosiy sinf yoki ota-sinfda mavjud usullarni qayta belgilashi mumkin. Bu erda bekor qilingan usul bir xil nomga ega, shuningdek o'tkazilgan argumentlarni anglatuvchi imzo va qaytish turiga ega.

23. Operatorning ortiqcha yuklanishi nima?

Operatorni haddan tashqari yuklash deganda u bilan birga berilgan argumentlar asosida foydalanuvchi tomonidan belgilangan turlardan foydalangan holda operatorlarni amalga oshirish tushuniladi.

24. Ortiqcha yuklanish va ortiqcha yuklanishni farqlang.

Overloading	Overriding
Two or more methods having the same name but different parameters or signature	Child class redefining methods present in the base class with the same parameters/ signature
Resolved during compile-time	Resolved during runtime

Inkapsulyatsiya

25. Inkapsulyatsiya nima?

Inkapsulyatsiya ma'lumotlar va ular ustida ishlaydigan kodni bitta birlikda birlashtirishni anglatadi. Masalan, sinf. Inkapsulyatsiya ma'lumotlarni yashirishga ham imkon beradi, chunki bitta sinfda ko'rsatilgan ma'lumotlar boshqa sinflardan yashiringan.

26. «Kirish spetsifikatsiyalari» nima?

Kirish spetsifikatorlari yoki kirish modifikatorlari OOPlarda usullar, sinflar va hokazolarning mavjudligini aniqlaydigan kalit so'zlardir. Ushbu kirish spetsifikatsiyalari inkapsulyatsiyani amalga oshirishga imkon beradi. Eng

keng tarqalgan kirish spetsifikatsiyalari umumiy, shaxsiy va himoyalangan. Biroq, dasturlash tillariga xos bo'lgan yana bir nechta mavjud.

27. Umumiy, xususiy va himoyalangan kirish modifikatorlarining farqi nimada?

Name	Accessibility from own class	Accessibility from derived class	Accessibility from world
Public	Yes	Yes	Yes
Private	Yes	No	No
Protected	Yes	Yes	No

Ma'lumotlarni abstraksiya qilish

28. Ma'lumotlarni abstraksiyalash nima?

Ma'lumotlarni abstraksiya qilish OOP ning juda muhim xususiyati bo'lib, u faqat muhim ma'lumotlarni ko'rsatish va amalga oshirish tafsilotlarini yashirish imkonini beradi. Misol uchun, velosiped haydash paytida siz gaz pedalini ko'tarsangiz, tezlik oshishini bilasiz, lekin bu qanday sodir bo'lishini bilmaysiz. Bu ma'lumotlarning abstraksiyasidir, chunki amalga oshirish tafsilotlari chavandozdan yashiringan.

29. Ma'lumotlarni abstraksiyalashga qanday erishish mumkin?

Ma'lumotlarning abstraksiyasiga quyidagilar orqali erishish mumkin:

- Abstrakt sinf
- Abstrakt usul

30. Abstrakt sinf nima?

Abstrakt sinf mavhum usullardan tashkil topgan sinfdir. Ushbu usullar asosan e'lon qilingan, ammo aniqlanmagan. Agar ushbu usullar ba'zi kichik

sinflarda qo'llanilishi kerak bo'lsa, ular faqat kichik sinfda aniqlanishi kerak.

31. Abstrakt sinfning misolini yarata olasizmi?

Yo'q. Mavhum sinfning namunalarini yaratib bo'lmaydi, chunki u to'liq amalga oshirilmagan. Biroq, mavhum sinfni meros qilib olgan pastki sinf misollari yaratilishi mumkin.

32. Interfeys deb nimaga aytiladi?

Bu metodlarni aniqlamasdan e'lon qilish imkonini beruvchi OOP tushunchasi. Interfeyslar, sinflardan farqli o'laroq, rejalar emas, chunki ular batafsil ko'rsatmalar yoki bajarilishi kerak bo'lgan harakatlarni o'z ichiga olmaydi. Interfeysni amalga oshiradigan har qanday sinf interfeys usullarini belgilaydi.

33. Ma'lumotlarni abstraksiyalash va inkapsulyatsiyani farqlang.

Data abstraction	Encapsulation
Solves the problem at the design level	Solves the problem at the implementation level
Allows showing important aspects while hiding implementation details	Binds code and data together into a single unit and hides it from the world

Usullari va vazifalari OOP intervyu savollari

34. Virtual funktsiyalar nima?

Virtual funktsiyalar - bu asosiy sinfda mavjud bo'lgan va pastki sinf tomonidan bekor qilingan funktsiyalar. Bu funktsiyalar ish vaqti polimorfizmga erishish uchun ishlatiladi.

35. Sof virtual funktsiyalar nima?

Sof virtual funktsiyalar yoki mavhum funktsiyalar faqat asosiy sinfda e'lon qilingan funktsiyalardir. Bu shuni anglatadiki, ular asosiy sinfda hech qanday ta'rifni o'z ichiga olmaydi va pastki sinfda qayta belgilanishi kerak.

36. Konstruktor nima?

Konstruktor - bu sinf bilan bir xil nomga ega bo'lgan va shu sinf ob'ektlarini ishga tushirish uchun ishlatiladigan maxsus turdagi usul.

37. Destruktor nima?

Destruktor - ob'ekt yo'q qilinganda avtomatik ravishda chaqiriladigan usul. Destruktor, shuningdek, yo'q qilingan ob'ektga ajratilgan yig'ish maydonini tiklaydi, ob'ektning fayllari va ma'lumotlar bazasi ulanishlarini yopadi va hokazo.

38. Konstruktorlarning turlari

Konstruktorlarning turlari tildan tilga farq qiladi. Biroq, barcha mumkin bo'lgan konstruktorlar:

- Standart konstruktor
- Parametrlangan konstruktor
- Konstruktorni nusxalash
- Statik konstruktor
- Xususiy konstruktor

39. Nusxalash konstruktori nima?

Nusxa ko'chirish konstruktori bir xil sinfning boshqa ob'ektdan o'zgaruvchilarni nusxalash orqali ob'ektlarni yaratadi. Nusxa ko'chirish konstruktorining asosiy maqsadi mavjud ob'ektdan yangi ob'ekt yaratishdir.

40. “Yakunlash” nima uchun ishlatiladi?

Boshqarilmaydigan resurslarni bo'shatish va Chiqindilarni yig'ish (GC) dan oldin tozalash uchun foydalaniladigan ob'ekt usuli sifatida yakunlang. U xotirani boshqarish vazifalarini bajaradi.

41. Garbage Collection (GC) nima?

GC - bu xotirani avtomatik boshqarishning amalga oshirilishi. Chiqindilarni yig'uvchi endi mavjud bo'lmagan ob'ektlar egallagan joyni bo'shatadi.

42. Sinf va metodni farqlang.

Class	Method
A class is basically a template that binds the code and data together into a single unit. Classes consist of methods, variables, etc	Callable set of instructions also called a procedure or function that are to be performed on the given data

43. Abstrakt sinf va interfeysni farqlang?

Basis for comparison	Abstract Class	Interface
Methods	Can have abstract as well as other methods	Only abstract methods
Final Variables	May contain final and non-final variables	Variables declared are final by default
Accessibility of Data Members	Can be private, public, etc	Public by default
Implementation	Can provide the implementation of an interface	Cannot provide the implementation of an abstract class

44. Yakuniy o'zgaruvchi nima?

Qiymati o'zgarmaydigan o'zgaruvchi. U har doim bir xil ob'ektga ko'ndalang bo'lmaganlik xususiyati bilan murojaat qiladi.

Istisno bilan ishlash

45. Istisno nima?

Istisno - bu dasturning normal bajarilishini to'xtatadigan xabarnomaning bir turi. Istisnolar xatoga naqsh beradi va uni hal qilish uchun xatoni istisno ishlovchisiga o'tkazadi. Istisno paydo bo'lishi bilan dasturning holati saqlanadi.

46. Istisnolarni qayta ishlash nima?

Ob'ektga yo'naltirilgan dasturlashda istisnolardan foydalanish xatolarni boshqarish uchun ishlatiladigan juda muhim tushunchadir. Istisno ishlov beruvchisi xatolarni tashlash va ushlash imkonini beradi va ularni hal qilish uchun markazlashtirilgan mexanizmni amalga oshiradi.

47. Xato va istisno o'rtasidagi farq nima?

Error	Exception
Errors are problems that should not be encountered by applications	Conditions that an application might try to catch

48. try/catch bloki nima?

Istisnolarni hal qilish uchun try/catch blokidan foydalaniladi. Try bloki xatoga olib kelishi mumkin bo'lgan bayonotlar to'plamini belgilaydi. Catch bloki asosan istisnoni ushlaydi.

49. Finally blok nima?

Nihoyat blok muhim kodni bajarish uchun ishlatiladigan koddan iborat, masalan, ulanishni yopish va hokazo. Bu blok try bloki chiqqanda bajariladi. Bundan tashqari, u kutilmagan istisnoga duch kelgan taqdirda ham blokning bajarilishini ta'minlaydi.

OOP cheklavlari

50. OOPning cheklavlari qanday?

- Odatda kichik muammolar uchun mos kelmaydi
- Intensiv sinovni talab qiladi
- Muammoni hal qilish uchun ko'proq vaqt talab etiladi
- To'g'ri rejalashtirishni talab qiladi
- Dasturchi muammoni ob'ektlar nuqtai nazaridan hal qilishni o'ylashi kerak

Umid qilamanki, siz ushbu qo'llanmada siz bilan baham ko'rilgan barcha narsalarni tushunasiz. Bu bizni OOPs intervyu savollari haqidagi maqolamizning oxiriga olib keladi. *Imkon qadar ko'proq mashq qilishingizga ishonch hosil qiling va tajribangizni qaytaring.*

Agar siz Sun'iy intellekt, DevOps, Ethical Hacking kabi bozorning eng ommabop texnologiyalari haqida ko'proq maqolalarni ko'rishni istasangiz, [Edureka rasmiy saytiga murojaat qilishingiz mumkin.](#)

Python va Data Sciencening boshqa jihatlarini tushuntirib beradigan ushbu turkumdagi boshqa maqolalarga e'tibor bering.

1. [Python-da Mashina o'rganish tasniflagichi](#)
2. [Python Scikit - Cheat Sheet-ni o'rganing](#)
3. [Mashinani o'rganish vositalari](#)
4. [Ma'lumotlar fanlari va mashinalarni o'rganish uchun Python kutubxonalari](#)
5. [Python tilidagi chatbot](#)

6. Python to'plamlari
7. Python modullari
8. Python dasturchi ko'nikmalari
9. Python yordamida veb-qirgish
10. Python dasturchisi uchun rezyume
11. Pythonda tadqiqot ma'lumotlarini tahlil qilish
12. Pythonning toshbaqa moduli bilan ilon o'yini
13. Python dasturchi ish haqi
14. Asosiy komponentlar tahlili
15. Python va C++
16. Scrapy o'quv qo'llanma
17. Python SciPy
18. Eng kichik kvadratlar regressiyasi usuli
19. Jupyter Notebook Cheat Sheet
20. Python asoslari

21. Python Pattern dasturlari
22. Python tilidagi generatorlar
23. Python dekoratori
24. Python Spyder IDE
25. Pythonda Kivy dan foydalanadigan mobil ilovalar
26. Python-ni o'rganish va mashq qilish uchun eng yaxshi 10 ta kitob
27. Python bilan robot ramkasi
28. PyGame yordamida Pythonda ilon o'yini
29. Django intervyu savollari va javoblari
30. Top 10 Python ilovalari
31. Python-da xash jadvallari va xashmaplar
32. Python 3.8
33. Vektor mashinasini qo'llab-quvvatlash
34. Python bo'yicha qo'llanma

• • •

Dastlab <https://www.edureka.co> saytida chop etilgan .

[Dasturlash](#)[Python](#)[Op tushunchalar](#)[Ha, tushunchalar](#)[Intervyu savollari](#)

Aayushi Johari tomonidan yozilgan

[Kuzatish](#)

946 obunachi · uchun yozuvchiEdureka

Turli texnologiyalar, jumladan Python, Data Science, Java va boshqalar haqida yozishni va bilimlarni tarqatishni yoqtiradigan texnologiya ishqibozi.

Aayushi Johari va Edurekadan ko'proq

ification Training



Real-life Projects



24x7 Tech Support



Aayushi Johari ichida Edureka

Misol bilan Python da massivlardan qanday foydalanishni bilib oling

Ushbu Python massivlari maqolasida massiv asoslari, masalan, funktsiyalar, ro'yxatlar va...

7 daqiqa o'qish · 5 aprel, 2019 yil



72



1



...

What is Dependency Injection?



Swatee Chand ichida Edureka

Spring Boot yordamida bog'liqlik kiritish

Dependency Injection haqidagi ushbu maqola Spring Boot-da amaliy qo'llash texnikasi...

11 daqiqa o'qish · 2019 yil 2 iyul



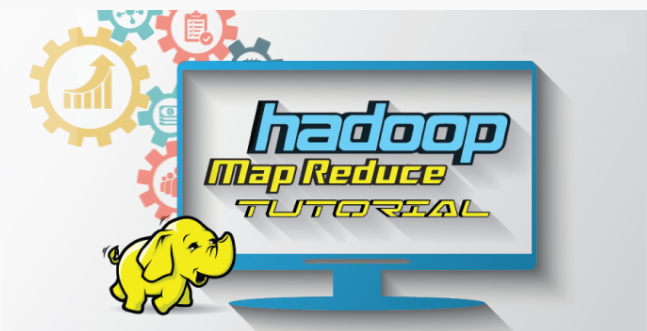
339



4



...



Shubham Sinha ichida Edureka

MapReduce misoli bilan MapReduce asoslari

Ushbu MapReduce qo'llanmasida siz MapReduce haqida hamma narsani bilib...

10 daqiqa o'qish · 2016 yil 15 noyabr



318



3



...



Aayushi Johari ichida Edureka

Python Pandas qo'llanmasi - Ma'lumotlarni tahlil qilish uchun...

Ushbu blog sizga Python Pandas ma'lumotlarni tahlil qilish va manipulyatsiya...

9 daqiqa o'qish · 2018 yil 5 aprel



39



1

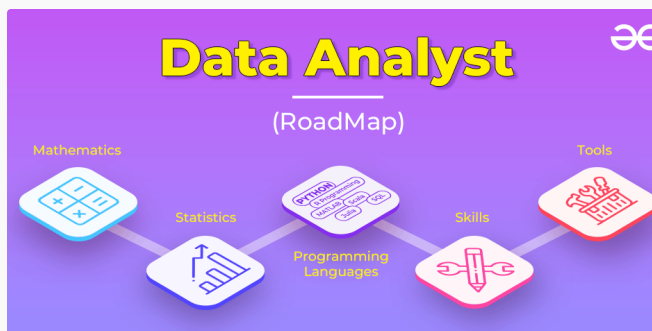


...

Hammasini Aayushi Joharidan ko'ring

Hammasini Edurekadan ko'ring

Mediumdan tavsiya etiladi



Afreen

Ma'lumotlar tahlilchisi bo'lish uchun 2023 yilgi yo'l xaritasi:...

Ma'lumotlar tahlili dunyosi tez sur'atlar bilan kengayib bormoqda va 2023 yilda...

★ · 3 daqiqa o'qish · 25 iyul

11



Aleksandr obidiegvu

Python ko'nikmalaringiz va portfelingizni oshirish uchun...

Har bir loyihani tugatish muddati - 1 oy

5 daqiqa o'qish · 3 kun oldin

1



Ro'yxatlar



Kodlash va ishlab chiqish

11 hikoya · 348ta tejash



Umumiy kodlash bilimi

20 hikoya · 727ta tejash



Python bilan bashoratli modellashtirish

20 hikoya · 734ta tejash



ChatGPT

23 hikoya · 357ta tejash



Python asoslari ichida Python oddiy ingliz tilida

Ma'lumot olimi intervyusi: 10 ta muhim savol va javob

Ma'lumot olimi bilan suhbatga tayyorgarlik juda qiyin vazifa bo'lishi mumkin. Ma'lumotla...

🌟 · 5 daqiqa o'qish · 11-sentabr



51



Divyanshi kulkarni

2024 yilda foydali martaba uchun eng yaxshi ma'lumotlar fanlari...

2024 yilda foydali martaba variantini qidirapsizmi? Yoki kelgusi yilda yanada...

5 daqiqa o'qish · 18-sentabr



3



Naim Azam

Java dasturchisi bo'lish uchun yo'l xaritasi: bosqichma-bosqich 4 yill...

1-yil: Java dasturlash asoslari



Hugo

Meta Android Developer Professional Coursera sertifikatig...

Men yaqinda Coursera-da [Meta Android Developer Professional] dasturini...

3 daqiqa o'qish · 27 iyul

2 daqiqa o'qish · 20-sentabr



Boshqa tavsiyalarni ko'ring