

PYCHARM MUHITI VA UNING DASTURLASHDAGI O'RNI

Tojimamatov Israiljon Nurmamatovich*Farg'ona davlat univarsiteti amaliy matematika va
informatika kafedrası katta o'qituvchisi
israiltojimatov@gmail.com***Solijonova Gulsanam Sodirjon qizi***Farg'ona davlat universiteti 3-kurs talabasi
gulsanamsolijonova18@gmail.com*

Annotatsiya: Ushbu maqolada PyCharm dasturlash muhiti va uning Python dasturlashidagi o'rni keng qamrovli tahlil qilinadi. PyCharm IDE ning tuzilishi, komponentlari, funksional imkoniyatlari, integratsiya mexanizmlari hamda dasturchilar samaradorligini oshirishdagi roli chuqur o'rganiladi. IDE ning kod tahlili, refaktoring vositalari, virtual muhitlar bilan ishlash imkoniyatlari, web-ramkalar bilan integratsiyasi va testlash tizimlari alohida ko'rib chiqiladi. Maqola ta'lim jarayonida, ilmiy ishlarda va real amaliy loyihalarda PyCharm platformasidan foydalanishning afzalliklarini ochib beradi.

Kalit so'zlar: PyCharm, IDE, Python, kod muharriri, refaktoring, xatolarni tuzatish, debugger, virtual muhit, Git integratsiyasi, web-ramkalar, Django, Flask, FastAPI, testlash, samaradorlik, ma'lumotlar bazasi integratsiyasi, dasturiy injiniring

Annotation: This article provides an extensive analysis of the PyCharm development environment and its role in Python programming. The structure, components, functionalities, integration capabilities, and productivity benefits of the PyCharm IDE are studied in depth. Special attention is given to code analysis tools, refactoring mechanisms, virtual environment management, integration with web frameworks, and testing systems. The article highlights the advantages of using PyCharm in educational processes, scientific research, and real-world software development projects.

Keywords: graph, node, search, time complexity, space complexity, efficiency, directed graph, shortest path. PyCharm, IDE, Python, code editor, refactoring, debugging, virtual environment, Git integration, web frameworks, Django, Flask, FastAPI, testing, productivity, database integration, software engineering.

Аннотация: В данной статье представлен углубленный анализ среды разработки PyCharm и её роли в программировании на Python. Подробно рассматриваются структура IDE, её компоненты, функциональные возможности, механизмы интеграции и влияние на повышение продуктивности разработчиков. Особое внимание уделяется инструментам анализа кода, механизмам рефакторинга, работе с виртуальными средами, интеграции с веб-фреймворками

и системам тестирования. Статья демонстрирует преимущества использования PyCharm в образовательных целях, научных исследованиях и реальных программных проектах.

Ключевые слова: PyCharm, IDE, Python, редактор кода, рефакторинг, отладка, виртуальная среда, интеграция Git, веб-фреймворки, Django, Flask, FastAPI, тестирование, продуктивность, интеграция базы данных, программная инженерия

Zamonaviy dasturlash jarayonida qulay, tezkor va funksional integratsiyalashgan rivojlantirish muhitlari (IDE) dasturchilar uchun eng asosiy vositalardan biriga aylangan. Python dasturlash tili so‘nggi yillarda butun dunyoda o‘zining kuchli imkoniyatlari, oson o‘rganilishi, ilmiy, amaliy va sanoat sohalaridagi keng qo‘llanilishi bilan yetakchi tillardan biriga aylandi. Pythonning ommalashuvi esa uning uchun maxsus ishlab chiqilgan dasturlash muhitlari rivojlanishiga sabab bo‘ldi. Shunday muhitlarning eng mashhuri va eng kuchlilaridan biri — PyCharm IDE hisoblanadi. JetBrains kompaniyasi tomonidan ishlab chiqilgan ushbu muhit bugungi kunda professional dasturchilar, ma‘lumotlar tahlilchilari, web-dasturchilar hamda ta‘lim sohasidagi mutaxassislar tomonidan keng qo‘llanilmoqda.

PyCharm o‘zining keng funksional imkoniyatlari orqali dasturlash jarayonining barcha bosqichlarini: kod yozish, uni tahlil qilish, debugging jarayonlari, testlash, refaktoring, virtual muhitlar bilan ishlash, versiya nazorati, web-ramkalar bilan integratsiya qilish va loyihalarni boshqarishni to‘liq qamrab oladi. IDE yordamida dasturchi bir vaqtning o‘zida katta loyihalar ustida ishlashi, murakkab kodlar orasida tez navigatsiya qilishi va samaradorlikni sezilarli darajada oshirishi mumkin.

PyCharmning asosiy afzalliklaridan biri — bu uning kuchli va aqlli kod muharriri hisoblanadi. Kod muharriri sintaksisni avtomatik ranglash orqali kodni vizual jihatdan qulay qiladi, murakkab tuzilmalarni ajratib ko‘rsatadi, kod tarkibidagi xatoliklarni real vaqt rejimida aniqlaydi va dasturchiga tahrirlash bo‘yicha takliflar beradi. Intellisense funksiyasi katta loyihalarda modullar va metodlarning nomlarini eslab qolish zaruratini kamaytiradi. PyCharm tomonidan taklif etiladigan avtomatik to‘ldirish, kodni formatlash, imlo xatolarni aniqlash kabi imkoniyatlar dasturchi vaqtini tejaydi va xatolarsiz kod yozishga xizmat qiladi. Kod tahlili PyCharmning eng kuchli elementlaridan biri bo‘lib, u kod tarkibini tahlil qiladi, ortiqcha kodlar, keraksiz importlar, nooptimal algoritmlar, qayta ishlatilmagan o‘zgaruvchilar va potentsial xatolarni aniqlaydi. PyCharm PEP-8 standartiga moslikni ham tekshiradi va kod sifati bo‘yicha maslahatlar beradi. Bu esa professional dasturlashdagi eng muhim omillardan biri — kod standartlari va intizomiga rioya qilishni osonlashtiradi.

PyCharmning yana bir o‘ta muhim funksiyasi — debugging vositalari hisoblanadi. Debugger dasturchiga dastur bajarilishini bosqichma-bosqich nazorat

qilish, o'zgaruvchilar qiymatini real vaqt rejimida tahlil qilish, xotira va jarayonlarni kuzatish imkoniyatini beradi. Breakpointlar yordamida kerakli joyda kod bajarilishini to'xtatish, murakkab xatolarni aniqlash, recursion jarayonlarini tahlil qilish juda qulay. Professional darajadagi debugging dasturchining vaqtini sezilarli tejaydi va xatolarni topish jarayonini ancha yengillashtiradi. PyCharmning virtual muhitlar bilan ishlash imkoniyatlari uning eng katta afzalliklaridan biridir. Python loyihalari odatda turli versiyadagi kutubxonalar, paketlar va modullardan foydalanadi. Agar paketlar bir-biriga mos kelmasa yoki global muhitga aralashsa loyihaning ishlashida muammolar vujudga kelishi mumkin. PyCharm esa virtualenv, venv, pipenv va conda kabi virtual muhitlarni avtomatik aniqlaydi, ularga yangi muhit yarata oladi, paketlarni boshqaradi va har bir loyiha uchun alohida muhit bilan ishlashni ta'minlaydi. Bu professional programmalashda juda muhim hisoblanadi.

PyCharmning o'ziga xos xususiyatlaridan yana biri — bu web-dasturlashni to'liq qo'llab-quvvatlashidir. Django, Flask, FastAPI, Pyramid kabi mashhur web-ramkalar PyCharm Professional versiyasida chuqur integratsiya qilingan. Django loyihalarida models, views, templates tuzilmalari avtomatik aniqlanadi, migrationlar boshqariladi, URL xaritalar bilan ishlash osonlashtiriladi. Web-loyihalarda PyCharmning serverni avtomatik ishga tushirish, templatelarda xatolarni aniqlash, JSON va API obyektlarini tahlil qilish imkoniyatlari web-dasturchilar uchun katta qulaylik yaratadi. PyCharm testlash tizimlarini ham keng qo'llab-quvvatlaydi. Pythonning unittest, pytest, nose2 kabi test paketlari uchun maxsus interfeyslar mavjud. Test natijalari grafik ko'rinishda ko'rsatiladi, muvaffaqiyatsiz testlar belgilab beriladi, loglar, xatolik izlari va koddagi aniq joylar ko'rsatib beriladi. Bu dasturiy injiniring jarayonida test-first yoki test-driven development metodologiyalarini qo'llashni sezilarli darajada soddalashtiradi.

Versiya nazorati (VCS) PyCharmda juda katta ahamiyat kasb etgan. Git, GitHub, GitLab, Bitbucket kabi tizimlar bilan integratsiya IDEga to'liq o'rnatilgan. Dasturchi IDE ichida commit, push, merge, rebase, diff ko'rish, branch yaratish kabi barcha amallarni bajara oladi. Grafik interfeys orqali kod o'zgarishlarini kuzatish, branchlar bilan ishlash, commit tarixini ko'rish juda qulay.

PyCharm data science sohasida ham keng qo'llaniladi. NumPy, pandas, Matplotlib, SciPy, TensorFlow, Keras, PyTorch kabi ilmiy kutubxonalarni to'liq qo'llab-quvvatlaydi. Jupyter Notebook fayllarini ochish, tahrirlash, bajarish imkoniyatlari ham mavjud. Data science bilan shug'ullanadigan mutaxassislar uchun PyCharmning profayllari va vizualizatsiya vositalari juda foydalidir.

Ta'lim jarayonida PyCharm o'quvchilar va talabalar uchun eng qulay vositalardan biri hisoblanadi. Uning intuitiv interfeysi, avtomatik tahlil va tushuntirish funksiyalari, kodning strukturaviy ko'rinishi, loyihalarni modellashtirish va tartibga keltirish osonligi Pythonni o'rganayotganlar uchun ideal sharoit yaratadi.

Umuman olganda, PyCharm zamonaviy dasturiy ta'minot ishlab chiqish jarayonida muhim o'rin egallagan. Uning ko'plab afzalliklari — qulay interfeys, kuchli debugging, kengaytirilgan refaktoring, testlash vositalari, virtual muhitlar bilan ishlash, web-ramkalar integratsiyasi, ma'lumotlar bazalari bilan ishlash va kuchli kod tahlili uni nafaqat professional dasturchilar, balki talaba, o'qituvchi va ilmiy tadqiqotchilar uchun ham eng yaxshi tanlovlardan biriga aylantiradi.

Foydalanilgan adabiyotlar:

1. Jordan, J. Intro to optimization in deep learning: Gradient Descent/ J. Jordan // Paper-space.Series:Optimization.— 2018. — URL: <https://blog.paperspace.com/intro-to-optimization-in-deep-learning-gradient-descent/>
2. Scikit-learn – машинное обучение на Python. — URL: http://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html
3. Keras documentation: optimizers. — URL: <https://keras.io/optimizers>
4. Ruder, S. An overview of gradient descent optimization algorithms / S. Ruder // Cornell University Library. — 2016. — URL: <https://arxiv.org/abs/1609.04747>
5. Robbins, H. A stochastic approximation method / H. Robbins, S. Monro // The annals of mathematical statistics. — 1951. — Vol. 22. — P. 400–407.
6. Kukar, M. Cost-Sensitive Learning with Neural Networks / M. Kukar, I. Kononenko // Machine Learning and Data Mining : proceedings of the 13th European Conference on Artificial Intelligence. — 1998. — P. 445–449.
7. Duchi, J. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization / J. Duchi, E. Hazan, Y. Singer // The Journal of Machine Learning Research. — 2011. — Vol. 12. — P. 2121–2159.
8. Zeiler, M. D. ADADELTA: An Adaptive Learning Rate Method / Cornell University Library. — 2012. — URL: <https://arxiv.org/abs/1212.5701>
9. Kingma, D. P. Adam: A Method for Stochastic Optimization / D. P. Kingma, J. Ba // Cornell University Library. — 2014. — URL: <https://arxiv.org/abs/1412.6980>
10. Гудфеллоу, Я. Глубокое обучение / Я. Гудфеллоу, И. Бенджио, А. Курвилль. — М. : ДМК Пресс, 2018. — 652 с.
11. Fletcher, R. Practical methods of optimization / R. Fletcher. — Wiley, 2000. — 450 p.
12. Schraudolph, N. N. A Stochastic Quasi-Newton Method for Online Convex Optimization / N.N. Schraudolph, J. Yu, S. Gunter // Statistical Machine Learning. — 2017.—URL: <http://proceedings.mlr.press/v2/schraudolph07a/schraudolph07a.pdf>
13. Ruder, S. Optimization for Deep Learning Highlights in 2017 / S. Ruder //

Optimization for Deep Learning Highlights in 2017. – 2017. – URL: <http://ruder.io/deep-learning-optimization-2017>.

14. Рахимов, К. (2022). Тенденции развития анализа данных, искусственного интеллекта и интернета вещей. *Gospodarka i Innowacje*, 30, 209-211.

15. Ortiqovich, Q. R., & Mamurovich, V. R. (2022). Mashinali o'qitish matematik modellari tasnifi. o'zbekistonda fanlararo innovatsiyalar va ilmiy tadqiqotlar jurnali, 2(14), 804-811.

16. Raximov, Q., & Sotvoldiyev, A. D. o'g'li. (2023). Neyron tarmoqlarining yangi turlarini tahlil qilish. international scientific and practical conference "the time of scientific progress ", 2(4), 106–112. Retrieved from <http://academicsresearch.ru/index.php/ispctosp/article/view/1500>

17. Ortiqovich, Q. R. (2022). Vorisli deformasiyalanuvchi konsruksiya-larning flatter masalasi. o'zbekistonda fanlararo innovatsiyalar va ilmiy tadqiqotlar jurnali, 2(14), 812-817.

18. Tojimamatov, I. N., Mamalatipov, O. M., & Karimova, N. A. (2022). SUN'IY NEYRON TARMOQLARINI O 'QITISH USULLARI. *Oriental renaissance: Innovative, educational, natural and social sciences*, 2(12), 191-203.

19. Tojimamatov, I. N., Mamalatipov, O., Rahmatjonov, M., & Farhodjonov, S. (2023). NEYRON TARMOQLAR. *Наука и инновация*, 1(1), 4-12.

20. Ne'matjonov, F. F., Jahongirova, J. J., Murodov, B. S., & Tojimamatov, I. N. (2023). CREATE DATA CUBE WITH MS EXCEL. *European International Journal of Multidisciplinary Research and Management Studies*, 3(03), 77-86.

21. Kimyonazarova, D., Ne'matjonova, D., Ergasheva, B., & Tojimamatov, I. (2023, March). KATTA MA'LUMOTLAR BILAN ISHLASHDA HADOOP ARXITEKTURASI. In *Международная конференция академических наук (Vol.2, No. 3, pp. 96-99)*.

22. Onarqulov, M., Yaqubjonov, A., & Yusupov, M. (2022). Computer networks and learning from them opportunities to use. *Models and methods in modern science*, 1(13), 59-62.

23. Karimberdiyevich, O. M., Mahamadamin o'g'li, Y. A., & Abdulaziz o'g'li, Y. M. (2023). MASHINALI O'QITISH ALGORITMLARI ASOSIDA BASHORAT QILISH USULLARINI YARATISH. *Journal of new century innovations*, 22(2), 165-167.

24. Mamirovich, I. S. (2022). Finlyandiya Va O 'zbekistonda Ta 'Lim Tizimini Sifatini Solishtirma Taxlili. *Miasto Przyszłości*, 29, 347-350..