Python jest językiem programowania ogólnego przeznaczenia, zorientowanym obiektowo, który kładzie przede wszystkim nacisk na czytelność kodu. Jest on stosunkowo łatwy do nauczenia się i przez to stał się jednym z najczęściej wybieranych przez deweloperów językiem programowania na świecie, zaraz za Javą i C. Sprawdza się on wszędzie tam, gdzie mamy do czynienia z dużymi zbiorami danych oraz koniecznością ich sprawnej analizy czy wizualizacji. Ponadto Python jest szczególnie dobrze przystosowany do wdrażania uczenia maszynowego na dużą skalę.

**Analiza SWOT**

|  |  |
| --- | --- |
| Mocne strony | Słabe strony |
| - Prosta i czytelna składnia, co ułatwia naukę i zrozumienie dla początkujących programistów.  - Duża społeczność i bogata biblioteka modułów, umożliwiająca szybkie rozwijanie aplikacji.  - Wieloplatformowość - działa na różnych systemach operacyjnych.  - Wydajność i skalowalność, szczególnie przy tworzeniu aplikacji internetowych i analizie danych.  - Popularność w dziedzinach sztucznej inteligencji, analizie danych i uczeniu maszynowym. | - Wydajność w niektórych zastosowaniach może być niższa w porównaniu do języków kompilowanych.  - Zarządzanie pamięcią jest automatyczne, co może prowadzić do nieefektywnego zarządzania zasobami w przypadku nieostrożnego programowania.  - Python może być mniej odpowiedni do zadań wymagających niskopoziomowego dostępu do sprzętu.  - Brak wsparcia dla wielowątkowości na poziomie wydajnościowym ze względu na GIL (Global Interpreter Lock). |
| Szanse | Zagrożenia |
| - Rosnące zapotrzebowanie na umiejętności programowania Pythona w różnych sektorach, takich jak analiza danych, sztuczna inteligencja i IoT.  - Ciągłe ulepszenia w języku i narzędziach deweloperskich wspierają rozwój aplikacji w Pythonie.  - Integracja z innymi językami programowania i technologiami, takimi jak C/C++, umożliwiająca tworzenie hybrydowych rozwiązań. | - Konkurencja ze strony innych języków programowania, szczególnie w obszarach, gdzie Python nie jest optymalny pod względem wydajności.  - Możliwe problemy z kompatybilnością w przypadku zmian w wersjach języka lub modułach.  - Ryzyko przesycenia rynku programistami Pythona, co może wpłynąć na poziom konkurencyjności na rynku pracy. |

Wnioski:

Pomimo pewnych ograniczeń, jego zalety zdecydowanie przeważają. Jednak aby utrzymać jego pozycję jako lidera w świecie programowania, konieczne jest ciągłe doskonalenie i adaptacja do zmieniających się warunków rynkowych i technologicznych.

|  |  |
| --- | --- |
| **Obaszar zastosowania** | **przykłady** |
| Tworzenie gier | PyGame |
| Automatyzacja procesów | poprzez napisanie skryptu, który będzie odpowiedzialny za testowanie, upload na serwer (w przypadku aplikacji webowych) czy statyczną analiza kodu |
| Bioinformatyka | Biopython |
| Matematyka | wykonywania obliczeń  tworzeniu wykresów i graficznej prezentacji danych  Numpy, sympy, ipython, matplotlib |
| Biznes, Sztuczna inteligencja | wspiera tworzenie sieci neuronowych  Tensor Flow, PyTorch |
| Tworzenie aplikacji WWW | Django  Flask |
| tworzenie aplikacji desktopowych i mobilnych | PyQt  Kivy |

Prócz tego:

* Analiza danych
* Robotyka
* Wizualizacja danych
* Uczenie maszynowe
* Zbieranie danych
* Testowanie stron