

# Narzędzia programistyczne – języki skryptowe

## Podział języków programowania

Przed wszystkim języki programowania dzielą się na dwie grupy – skryptowe oraz kompilowane.

Pierwsza grupa to **języki skryptowe**. Programy napisane w ten sposób to właściwie pliki tekstowe (i tak, można je edytować nawet w notatniku), uruchamia się je za pomocą dodatkowego programu i w momencie uruchomienia tekst (kod) jest interpretowany (zamieniany na postać zrozumiałą dla komputera i wykonywany). Zaletą tego podejścia jest to, że możemy szybko zobaczyć efekty zmian (po zmianie wystarczy odświeżyć stronę czy uruchomić ponownie i od razu widzimy zmiany). Minusem jest, że błędy o bledach w kodzie (np. w składni) dowiemy się dopiero w momencie uruchamiania tego konkretnego fragmentu kodu (najczęściej), co utrudnia szukanie błędów. Drugim minusem, który jednak nie będzie problemem w większości przypadków, jest mniejsza wydajność takich aplikacji. W ogólnym przypadku jednak w językach skryptowych pisze się szybciej, ale lepiej sprawdzają się w prostszych aplikacjach/projektach.

Druga grupa to **języki kompilowane**, do tej grupy należy m.in. Java. Różnią się one tym, że zanim uruchomimy nasz kod, musimy go skompilować, tzn. przetłumaczyć na język zrozumiały dla komputera. W przypadku języków skryptowych proces ten odbywa się w momencie uruchamiania, za każdym razem jak program jest uruchamiany. W językach kompilowanych proces ten jest wykonywany tylko raz, jeszcze przed uruchomieniem programu. Powoduje to, że wprowadzone przez nas zmiany musimy najpierw skompilować (najczęściej z całym programem) a następnie uruchomić program jeszcze raz. Wymaga to trochę innego podejścia do pisania programów (metoda prób i błędów do rozwiązania problemów nie sprawdzi się w większych projektach). Z drugiej strony programy napisane z użyciem języków kompilowanych działają szybciej, a błędy w składni są wykrywane jeszcze przed uruchomieniem programu (na etapie kompilacji). Znajomość języków z tej grupy daje też większe możliwości na rynku pracy, ponieważ większość ofert z dużych firm dotyczy właśnie tej grupy.

W tym miejscu należy podkreślić, że **jedna grupa nie jest lepsza od drugiej**. Są inne, nadają się do innych rzeczy i rozwijając się w programowaniu czy kiedyś może zaczynając własny start-up należy rozważyć wady i zalety obu, nie odrzucając żadnego z góry.

## Podsumowując:

Języki skryptowe		Języki kompilowane	
Zalety	Wady	Zalety	Wady
<ul style="list-style-type: none"><li>• w ogólnym przypadku szybciej pisze się proste aplikacje</li><li>• najczęściej mniej ścisła konwencja, składnia więcej 'wybacza'</li><li>• pozwala na łatwe poprawianie błędów metodą prób i błędów, co jest pomocne przy nauce</li></ul>	<ul style="list-style-type: none"><li>• duże projekty są trudne w utrzymaniu i rozwijaniu w przyszłości</li><li>• mniej wydajny, co stanowi problem jak projekt zyskuje popularność</li><li>• języki najczęściej pozwalają na mieszanie kodu z elementami widoków co często prowadzi do nieczytelnego kodu</li></ul>	<ul style="list-style-type: none"><li>• wydajne, szybkie, łatwiej skalowalne (w przyszłości)</li><li>• dobrze napisany kod jest przeważnie łatwiejszy w utrzymaniu i rozwijaniu w przyszłości</li><li>• sam język wspiera czystszy kod i lepszą separację kodu od widoków</li></ul>	<ul style="list-style-type: none"><li>• napisanie aplikacji wymaga przeważnie więcej pracy niż z wykorzystaniem języków skryptowych</li><li>• przed uruchomieniem wymaga kompilacji, co w dużych projektach wydłuża proces znajdowania i poprawiania błędów</li></ul>

## Języki skryptowe

### PHP

PHP to jeden z najpopularniejszych języków skryptowych, także jeden z pierwszych powszechnie używanych. Powstał jako rozszerzenie popularnego kiedyś języka skryptowego Perl z myślą o aplikacjach internetowych. Nadal jest z nimi ściśle związany.

Język ten jest bardzo prosty do nauki podstaw, jednak przez bardzo dynamiczny rozwój i wiele zmian wprowadzanych bez głębszej refleksji, niektóre jego elementy zdają się chaotyczne, konwencje nazewnictwa czy argumentów nie wszędzie są zachowane i bywa to uciążliwe. Z drugiej strony bardzo łatwo o hosting aplikacji napisanych w PHP (prawie każdy serwer, także np na home.pl czy nazwa.pl obsługuje ten język).

Język bardzo popularny w Internecie, np. jako fora internetowe, portale, blogi itp

## Python / Django

Język zdobywający na popularności, w szczególności w połączeniu z frameworkiem Django (używanym do aplikacji webowych).

Ogromną zaletą tego języka jest jego prostota, prawdopodobnie jest to najłatwiejszy do nauczenia się język. Pewnym ograniczeniem jest hosting aplikacji w Pythonie (czyli jeśli chcemy udostępnić naszą aplikację światu). Nie ma też zbyt wielu (w porównaniu do innych technologii) ofert pracy dla programistów tej technologii.

Język ten jest używany głównie w start-upach oraz małych firmach.

## Ruby on Rails

Kolejny język/framework zdobywający popularność. Wyróżnikiem jest sam język Ruby, który powstał poprzez próbę połączenia najlepszych cech innych języków. Przewaga nad np. Pythonem sprowadza się głównie do preferencji osobistych lub specyficznych funkcji, niemniej jest to najmniej popularny język jeśli chodzi o oferty pracy.

Technologia ta jest używana głównie w start-upach oraz przez pasjonatów.

## Node.JS

Technologia o której ostatnio robi się głośno, cechą unikalną jest inne podejście do programowania aplikacji webowych, tzw. programowanie zdarzeniowe. Ma ono swoje zalety (dla zainteresowanych i nieco obeznanych w temacie). Jest też możliwe do realizacji w innych językach programowania (choć głównie kompilowanych), ale wymaga od programisty dość dużej wiedzy aby w pełni rozumieć co się dzieje w aplikacji. Raczej nie polecany dla początkujących, ale mimo tego warty uwagi.

Technologia ta obecnie jest najczęściej wybierana przez startupy lub w dużych firmach do projektów eksperymentalnych, nie kluczowych.

## JavaScript

JavaScript jest językiem skryptowym dla dokumentów internetowych. Skrypty napisane za pomocą JavaScript mogą być umieszczane bezpośrednio na stronach HTML. Język daje obszerne możliwości wzbogacania stron w interesujące elementy. Na przykład możesz zdefiniować reakcję na zainicjowane przez czytelnika działania. Niektóre efekty jeszcze niedawno były możliwe jedynie przy użyciu CGI, a teraz jesteś w stanie zbudować wyrafinowane konstrukcyjnie strony za pomocą JavaScript. Wiele przykładów użycia języka możesz zobaczyć w Internecie.

## Jaka jest różnica między językami Java i JavaScript

Mimo zbieżności nazw Java nie jest tożsama z JavaScript. Są to dwie odrębne techniki programowania w Internecie. Java jest językiem programowania, natomiast JavaScript jest językiem skryptowym, jak zresztą sugeruje sama nazwa. Różnica polega na tym, że Java pozwala tworzyć rzeczywiste programy. Ale nie zawsze musisz sobie zaprzętać głowę prawdziwym programowaniem, aby uzyskać ładny efekt na swojej stronie. JavaScript jest pomyślany jako narzędzie łatwe do zrozumienia i zastosowania. Autorów piszących w JavaScript nie musi zbytnio obchodzić programowanie. Można powiedzieć, że JavaScript jest raczej rozszerzeniem HTML niż odrębnym językiem komputerowym. Oczywiście nie jest to "oficjalna" definicja, ale myślę, że pozwala ona łatwiej uchwycić różnicę między Javą i JavaScript.

## Jak można uruchamiać skrypty JavaScript

Pierwszą przeglądarką obsługującą JavaScript był Netscape Navigator 2.0, jak i, oczywiście, późniejsze jego wersje. Być może wiesz, że nie wszystkie wersje Navigatora czytają Javę, ale JavaScript jest interpretowany na wszystkich platformach, choć występują pewne problemy w niektórych wersjach. Wersja dla Macintosha, dla przykładu, wydaje się mieć pewną ilość błędów. Naturalnie powinieneś już dysponować pewną znajomością języka HTML.

## Przykłady

Kilka skryptów, które zilustrują sposób ich implementowania na stronie HTML i możliwości oferowane przez ten język. Rozpocznijmy od skryptu, który będzie po prostu wyświetlał jakiś tekst.

```
<html>
<head>
My first JavaScript!
</head>
<body>
<br>
To jest normalny dokument HTML.
<br>
  <script language="JavaScript">
    document.write("To jest JavaScript!")
  </script>
<br>
I znowu dokument HTML.
</body>
</html>
```

Efekt po wykonaniu skryptu:

*To jest normalny dokument HTML.*

*To jest JavaScript!*

*I znowu dokument HTML.*

## Edytory tekstu dla programistów (wybór – tylko bezpłatne)

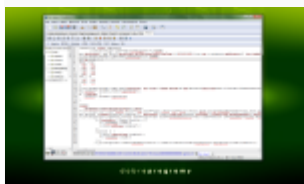
---

### Atom



Licencja: bezpłatna Atom to nowoczesny i uniwersalny edytor tekstu dla programistów, który udostępniony został bezpłatnie. Charakteryzuje się on przejrzystym i dopracowanym interfejsem graficznym, możliwością personalizacji wielu atrybutów jego działania oraz wyglądu

### Bluefish



Licencja: bezpłatna Bluefish to szybki i darmowy edytor tekstu przeznaczony dla programistów i webmasterów. Charakteryzuje się bogatą funkcjonalnością i szybkością działania. Obsługuje kolorowanie i podświetlanie składni dla wielu popularnych języków programowania.

Program umożliwia tworzenie własnych pasków narzędzi i generowanie różnych galerii zdjęć. Obsługuje również niemal wszystkie najpopularniejsze języki do programowania i projektowania witryn internetowych. Wśród nich warto wymienić między innymi: ASP .NET, C/C++, CSS, CFML, HTML, XHTML, HTML5, Java, JSP, JavaScript, jQuery, Lua, Pascal, Perl, PHP, Python, Ruby, SQL i XML.

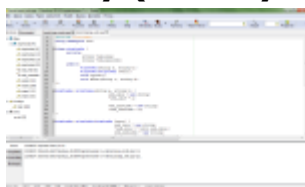
---

## Editra



**Licencja: bezpłatna** Editra to darmowy i wieloplatformowy edytor tekstu, stworzony z myślą o programistach i webmasterach. Program ten oferuje bogaty zestaw funkcji, pozwalających wygodnie tworzyć i edytować kod stron internetowych oraz aplikacji w wielu. Aplikacja pomimo swojej rozbudowanej funkcjonalności charakteryzuje się dużą szybkością działania. Z powodzeniem wykorzystywana ona może być do pracy nad prostymi, jak również bardziej złożonymi projektami stron WWW czy oprogramowania. Do dyspozycji użytkownika oddano edytor wyposażony m.in. w mechanizm kart, umożliwiający pracę z wieloma plikami jednocześnie, funkcję numerowania linii, przydatną podczas edycji bardziej rozbudowanych projektów, a także kolorowania składni kodu w 60-ciu najpopularniejszych językach.

## Geany (idealny dla Pythona)



**Licencja: bezpłatna** Geany to idealne połączenie edytora tekstu z prostym środowiskiem programistycznym, które cechuje się niezwykłą lekkością, szybkością i dopracowanym interfejsem graficznym. Narzędzie chociaż na pierwszy rzut oka wydawać by się mogło rozbudowanym edytorem to bardziej niż do redagowania tekstów nadaje się jako narzędzie pracy dla programistów i webmasterów. Geany oferuje kolorowanie składni dla najpopularniejszych języków - m.in. C, C++, C#, Java, PHP, HTML, Python, Perl i Pascal, wsparcie dla kodowania w ponad 50 standardach, mechanizm automatycznego zamykania tagów dla HTML/XML, auto-wcięcie, pracy na kartach i wiele, wiele więcej.

## Notepad++



**Licencja: bezpłatna** Notepad++ jest bardzo dobrym, darmowym edytorem kodów źródłowych. Projekt bazuje na potężnym komponencie edytora tekstu - Scintilla. Aplikacja jest napisana w C++, czystym Win32 Api i STL dzięki czemu działa wydajnie.

## Hosting stron WWW /bezpłatny i bez reklam/

- <https://www.hostinger.pl/>
- <https://www.cba.pl/>
- [www.prv.pl/](http://www.prv.pl/)
- <https://pl.000webhost.com>
- <https://www.friko.pl/>
- <https://5v.pl/>
- <https://www.ugu.pl/>

## Zadanie

1. Uruchom edytor Notepad++. Zapoznaj się z obsługą programu. Zapisz pliki z rozszerzeniem js, php, html.

## Praca domowa

1. Utwórz bezpłatne konto strony WWW. Zapamiętaj dane dostępowe w postaci nazwa strony (adres WWW strony) oraz dane dostępowe.
2. Zaprojektuj i utwórz stronę WWW (plik index.html). Strona powinna zawierać:
  - Logo (grafika typu baner)
  - Menu strony /poziome albo pionowe/
  - Miejsce na treść strony

Przykładowy szablon strony:

