

Kod źródłowy przykładów prezentowanych w tej książce występuje w postaci zarówno krótkich urywków, jak i kompletnych skryptów; zebraliśmy go w dwojakiej formie: klasycznych plików **.py* oraz notatników programu Jupyter, czyli plików **.ipynb*. Aby ułatwić pracę zarówno użytkownikom klasycznych edytorów dla Windows, jak i użytkownikom zaawansowanych narzędzi dla języka Python, podzieliliśmy udostępniany kod na dwa drzewa folderów, różniące się kodowaniem polskich liter w plikach **.py*:

- w drzewie folderów *CP1250* polskie litery zapisywane są zgodnie ze stroną kodową 1250, domyślną dla polskiej wersji MS Windows. Aby były poprawnie rozpoznawane przez interpretery języka Python, konieczne było umieszczenie we wspomnianych plikach instrukcji

```
#-*- coding: 1250 -*-
```

w pierwszej albo drugiej linii.
- w drzewie folderów *UTF8* polskie litery kodowane są w plikach **.py* wg standardu UTF-8, domyślnego dla języka Python i jego interpreterów. Każdy z tych plików rozpoczyna się od znacznika BOM identyfikującego wspomniany standard (szesnastkowo — EF BB BF), jawne wskazywanie kodowania za pomocą instrukcji `coding` nie jest konieczne.

Opisana różnica nie dotyczy plików **.ipynb*, które w obu drzewach mają identyczną postać — dokumentów JSON kodowanych w standardzie UTF-8. Są one przeznaczone do edytowania za pomocą webowego edytora *Jupyter Lab*, nie za pomocą niezależnych edytorów tekstowych. Także wszystkie pozostałe pliki, w szczególności pliki **.pdf* zawierające dodatkowe informacje, są w obu drzewach identyczne. Zdecydowaliśmy się na takie dublowanie, by oba drzewa były kompletne, wzajemnie niezależne i użytkownik mógł ograniczyć się do jednego, wygodniejszego dla siebie.

W kilku skryptach dla rozdziału 16., przeznaczonych do wykonywania na zewnętrznych klastrach Hadoop i Spark, dla uniknięcia ewentualnych problemów zrezygnowaliśmy z używania polskich liter, zastępując je odpowiednikami z kodu ASCII.

Narzędziem do uruchamiania przykładów jest darmowa dystrybucja języka Python o nazwie *Anaconda*, w jej ramach uruchamiany jest interpreter IPython i rozszerzenie notatników Jupytera integrujące je z tymże interpreterem. Użytkownik powinien więc rozpocząć przygodę z książką od zainstalowania wspomnianej dystrybucji; instalatory dla systemów Windows, Linux i MacOS dostępne są do pobrania pod adresem

<https://www.anaconda.com/products/individual>

(należy pobrać odpowiedni pakiet dla wersji 3.7 Pythona). Po uruchomieniu pobranego pakietu instalacja przebiegnie automatycznie, należy odpowiadać twierdząco na pytania systemu o kontynuację przetwarzania.

Dystrybucja Anaconda instalowana jest wraz z bogatym zestawem gotowych bibliotek z różnych dziedzin tematycznych, mimo to na potrzeby niektórych przykładów konieczne będzie doinstalowanie kilku innych komponentów. Scenariusze instalacji opisywane są ze szczegółami w treści odnośnych rozdziałów, tutaj chcemy jedynie zwrócić uwagę użytkowników Windows na pewien szczegół: otóż ze względu na specyfikę tego systemu, doinstalowując nowe komponenty, należy uruchomić konsolę *Anaconda prompt* w trybie administracyjnym (za pomocą opcji *Uruchom jako administrator*), w przeciwnym razie konsola (jako aplikacja) nie uzyska uprawnień dostępu do niektórych części systemu i instalacja może się załamać.

Niektóre przykłady wiążą się z wykorzystywaniem zewnętrznych usług, w większości darmowych lub kredytowanych — między innymi API Twittera, platformy IBM Watson czy klasterów Hadoop i Spark. Uzyskane w procesie rejestracji dane („klucze”) warunkujące dostęp do wspomnianych usług należy każdorazowo umieścić w odpowiednim pliku *keys.py*, w sposób opisywany w treści rozdziałów dla konkretnych usług; w dostarczanej przez nas wersji pliki *keys.py* są jedynie szablonami pozbawionymi konkretnych danych, przeznaczonymi do uzupełnienia w taki właśnie sposób..

Pamiętajmy, że kraina Pythona jest częścią uniwersum *open-source*, nieustannie rozwijanego przez jego społeczność: wciąż opracowywane są nowe rozwiązania i nowsze wersje istniejących, usuwane są zauważone błędy (i, niestety, popełniane są nowe), warto zatem regularnie weryfikować aktualność zarówno samej dystrybucji *Anaconda*, jak i zainstalowanych w jej środowisku bibliotek. Temu celowi służą dwa następujące polecenia:

```
conda update conda
conda update --all
```

uruchamiane z konsoli Anacondy (która w Windows powinna być uruchomiona w trybie administracyjnym).