

Spis treści

Przedmowa 15

Część I. Tworzenie interfejsów API do wykorzystania w danologii 21

1. Tworzenie API, które pokochają danolodzy 23

W jaki sposób danolodzy korzystają z API? 23

Jakich narzędzi używają danolodzy? 24

Projektowanie interfejsów API dla danologów 24

Wprowadzenie do projektu portfolio z części I 26

Każdy interfejs API ma swoją historię 26

Poznaj swoją firmę: SportsWorldCentral 26

SWC potrzebuje interfejsu API 29

Wybór pierwszych produktów API 30

Identyfikacja potencjalnych użytkowników 30

Tworzenie historyjek użytkownika 31

Dodatkowe zasoby 32

Podsumowanie 32

2. Wybór architektury API 34

Style architektoniczne interfejsu API 34

REST 35

GraphQL 35

gRPC 36

Twój wybór: REST 36

Architektura techniczna 38

Oprogramowanie używane w tym rozdziale 40

Python 40

GitHub 40 Pierwsze kroki z GitHub Codespaces 41

Tworzenie konta w serwisie GitHub 41

Klonowanie repozytorium części I 41

Uruchamianie środowiska GitHub Codespaces 42

Przeglądanie nowej przestrzeni kodu 42

Tworzenie pierwszego commita	43
Dodatkowe zasoby	46
Podsumowanie	46
3. Utworzenie bazy danych	47
Składniki Twojego API	47
Oprogramowanie używane w tym rozdziale	48
SQLite	48
SQLAlchemy	49
pytest	49
Tworzenie bazy danych SQLite	50
Tworzenie tabel w bazie danych	50
Struktura tabel	52
Wczytywanie danych	53
Dostęp do danych w Pythonie	54
Instalowanie SQLAlchemy w swoim środowisku	55
Tworzenie plików Pythona na potrzeby obsługi dostępu do bazy danych	56
Tworzenie pliku konfiguracyjnego bazy danych	60
Tworzenie funkcji pomocniczych SQLAlchemy	61
Instalacja pytest	65
Testowanie kodu SQLAlchemy	65
Dodatkowe zasoby	68
Podsumowanie	69
4. Tworzenie kodu FastAPI	70
Rozbudowa projektu portfolio	70
Oprogramowanie używane w tym rozdziale	71
FastAPI	71
HTTPX	72
Pydantic	72
Uvicorn	73
Kopiowanie plików z rozdziału 3.	73
Instalowanie nowych bibliotek w środowisku programistycznym	74

Tworzenie plików Pythona dla interfejsu API	74
Tworzenie schematów Pydantic	74
Tworzenie kontrolera FastAPI	78
Testowanie interfejsu API	84
Uruchamianie interfejsu API	87
Dodatkowe zasoby	89
Podsumowanie	89
5. Dokumentowanie interfejsu API	91
Wysyłanie sygnału zaufania	91
Tworzenie świetnych dokumentacji interfejsów API	92
Elementy kluczowe	92
Elementy dodatkowe	93
Przeglądanie przykładów dokumentacji API	94
Aplikacja Sleeper	94
MyFantasyLeague	95
Yahoo! Fantasy Football	96
Przeglądanie wbudowanej dokumentacji API	97
Kopiowanie plików z rozdziału 4.	97
Opcja dokumentacji 1. Swagger UI	98
Opcja dokumentacji 2. Redoc	104
Praca ze specyfikacją OpenAPI	105
Rozbudowa projektu portfolio	107
Dodawanie szczegółów do obiektu info w OAS	108
Dodawanie tagów do kategoryzacji ścieżek	109
Rozszerzanie informacji o poszczególnych punktach końcowych	110
Dodawanie opisów parametrów	110
Wyświetlanie zmian w interfejsie Swagger UI	111
Testy regresji API	112
Aktualizacja pliku README.md	112
Dodatkowe zasoby	114
Podsumowanie	115

6. Wdrażanie interfejsu API w chmurze 116

Korzyści i obowiązki związane z wdrażaniem rozwiązań chmurowych 116

Korzyści 116

Obowiązki 117

Wybór dostawcy chmury dla Twojego projektu 118

Konfigurowanie katalogu projektów 119

Korzystanie z GitHub Codespaces jako hosta w chmurze 119

Wdrażanie w usłudze Render 120

Rejestracja w usłudze Render 121

Tworzenie nowej usługi sieciowej 121

Automatyczne wdrażanie zmian w interfejsie API 123

Wdrażanie aplikacji w kontenerze Docker 124

Weryfikowanie instalacji platformy Docker 124

Tworzenie pliku Dockerfile 125

Tworzenie pliku .dockerignore 126

Tworzenie obrazu kontenera 126

Uruchamianie obrazu kontenera lokalnie 127

Wdrażanie do chmury AWS 128

Tworzenie usługi kontenerowej Lightsail 128

Instalowanie interfejsu wiersza poleceń platformy AWS 129

Instalacja wtyczki Amazon Lightsail Container Services 129

Konfiguracja danych logowania 129

Przesyłanie obrazu kontenera do usługi Lightsail 130

Tworzenie wdrożenia w Lightsail 131

Aktualizowanie dokumentacji API 133

Dodatkowe zasoby 134

Podsumowanie 134

7. Baterie w zestawie: tworzenie zestawu SDK języka Python 135

SDK niwelują różnice 136

Wybór języka dla Twojego SDK 139

Zacznij od minimalnej wersji pakietu SDK 140

Wskazówka eksperta. Jak ułatwić instalację pakietu SDK? 140

Wskazówka eksperta. Jak zapewnić spójność i idiomatyczność pakietu SDK? 141

Tworzenie zaawansowanego zestawu SDK 144

Wskazówka eksperta. Używanie rozsądnych ustawień domyślnych 144

Wskazówka eksperta. Zapewnianie rozbudowanej funkcjonalności 146

Wskazówka eksperta. Rejestrowanie zdarzeń 150

Wskazówka eksperta. Ukrywanie skomplikowanych szczegółów interfejsu API 153

Wskazówka eksperta. Obsługa pobierania w trybie masowym 154

Wskazówka eksperta. Dokumentowanie SDK 157

Testowanie pakietu SDK 159

Wskazówka eksperta. Implementacja wszystkich zadań obsługiwanych przez API 162

Dokończenie projektu portfolio w części I 163

Dodatkowe zasoby 165

Podsumowanie 166

Część II. Wykorzystanie API w projektach Data Science 167

8. Co o interfejsach API powinien wiedzieć danolog? 169

Korzystanie z różnych stylów interfejsu API 169

Podstawy protokołu HTTP 171

Jak odpowiedzialnie korzystać z interfejsów API? 172

Podział odpowiedzialności. Korzystanie z SDK lub tworzenie klientów API 174

Jak tworzyć interfejsy API? 176

Jak testować interfejsy API? 176

Wdrażanie i konteneryzacja interfejsów API 177

Stosowanie kontroli wersji 177

Wprowadzenie do projektu portfolio — część II 178

Pierwsze kroki z GitHub Codespaces 178

Klonowanie repozytorium części II 178

Uruchamianie środowiska GitHub Codespaces 179

Lokalne uruchomienie API SportsWorldCentral (SWC) 180

Dodatkowe zasoby 181

Podsumowanie 182

9. Korzystanie z interfejsów API do analizy danych 183

Niestandardowe metryki do analityki sportowej 183

Wykorzystanie interfejsów API jako źródeł danych

do tworzenia niestandardowych metryk w ligach fantasy 184

Tworzenie niestandardowej metryki. Shark League Score 186

Oprogramowanie wykorzystane w tym rozdziale 187

httplib 187

Notatniki Jupyter 187

pandas 188

Instalowanie nowych bibliotek w środowisku Github Codespaces 188

Uruchamianie interfejsu API w środowisku Codespaces 188

Tworzenie pliku klienta interfejsu API 189

Tworzenie notatnika Jupyter 190

Dodawanie ogólnej konfiguracji do notatnika 191

Praca z danymi API 193

Obliczanie współczynnika zrównoważenia ligi 195

Obliczanie współczynnika soczystości ligi 196

Obliczanie współczynnika ligi rekinów (SLS) 197

Dodatkowe zasoby 198

Podsumowanie 199

10. Używanie interfejsów API w potokach danych 200

Typy źródeł danych dla potoków przetwarzania danych 201

Planowanie potoku danych 201

Orkiestracja potoku danych z użyciem Apache Airflow 202

Instalacja Apache Airflow w środowisku GitHub Codespaces 203

Tworzenie lokalnej analitycznej bazy danych 207

Uruchamianie interfejsu API w środowisku Codespaces 208

Konfiguracja połączeń Airflow 208

Tworzenie pierwszego grafu DAG 208

Tworzenie funkcji wspólnej 212

Uruchamianie grafu DAG 213

Podsumowanie 215

11. Wykorzystanie API w aplikacjach danych Streamlit 216

Angażowanie użytkowników za pomocą interaktywnych wizualizacji 216

Oprogramowanie wykorzystane w tym rozdziale 217

nfl_data_py 218

Streamlit 218

Instalacja Streamlit i nfl_data_py 218

Uruchamianie interfejsu API w Codespaces 218

Ponowne wykorzystanie pliku klienta API z rozdziału 9. 219

Tworzenie aplikacji Streamlit 219

Aktualizacja pliku startowego 220

Uruchamianie aplikacji Streamlit 220

Tworzenie strony ze składami drużyn 221

Tworzenie strony ze statystykami zespołów 224

Wdrażanie aplikacji Streamlit 227

Uzupełnianie projektu portfolio — część II 228

Dodatkowe zasoby 229

Podsumowanie 229

Część III. Wykorzystanie interfejsów API w sztucznej inteligencji 231

12. Wykorzystanie API w sztucznej inteligencji 233

Połączenie sztucznej inteligencji i interfejsów API 233

Projektowanie interfejsów API do wykorzystania z generatywną sztuczną inteligencją i modelami językowymi 235

Czym jest sztuczna inteligencja? 237

Generatywna sztuczna inteligencja i duże modele językowe 238

Tworzenie aplikacji wykorzystujących inteligentne agenty AI 238

Wprowadzenie do projektu portfolio — część III 240

Pierwsze kroki z GitHub Codespaces 240

Klonowanie repozytorium części III 240

Uruchamianie środowiska GitHub Codespaces 241

Dodatkowe zasoby 242

Podsumowanie 242

13. Wdrażanie interfejsu API na potrzeby uczenia maszynowego 243

Szkolenie modeli uczenia maszynowego 244

Oprogramowanie wykorzystane w tym rozdziale 245

Środowisko wykonawcze ONNX 246

scikit-learn 246

sklearn-onnx 246

Instalowanie nowych bibliotek w środowisku Codespaces 246

Zastosowanie procesu CRISP-DM 247

Zrozumienie potrzeb biznesowych 248

Zrozumienie danych 248

Przygotowanie danych 250

Modelowanie 250

Ocena 253

Wdrażanie 253

Dodatkowe zasoby 260

Podsumowanie 261

14. Korzystanie z interfejsów API z LangChain 262

Wykorzystanie sztucznej inteligencji poprzez API (za pomocą LangChain) 263

Tworzenie agenta LangGraph 264

Rejestracja w usłudze Anthropic 265

Uruchamianie środowiska GitHub Codespaces 266

Instalowanie nowych bibliotek w środowisku GitHub Codespaces 267

Tworzenie notatnika Jupyter 267

Czat z agentem LangGraph 270

Uruchomienie API SportsWorldCentral (SWC) lokalnie 272

Instalacja pakietu SDK biblioteki swcipy 272

Tworzenie zestawu narzędzi LangChain 273

Wywoływanie API z użyciem sztucznej inteligencji (z LangGraph) 276

Rozmowa z Twoim agentem (z użyciem narzędzi) 278

Dodatkowe zasoby 280

Podsumowanie 280

15. Wykorzystanie ChatGPT do wywoływania API 281

Architektura aplikacji 281

Pierwsze kroki z ChatGPT 282

Tworzenie niestandardowego modelu GPT 282

Uruchamianie środowiska GitHub Codespace 286

Uruchamianie interfejsu API SportsWorldCentral (SWC)

w środowisku GitHub Codespace 286

Dodawanie opisu serwerów do pliku OAS 287

Tworzenie działania GPT 288

Testowanie interfejsów API w Twoim modelu GPT 290

Rozmowy z własnym GPT 292

Uzupełnianie projektu portfolio części III 294

Podsumowanie 295