## Cel zadania

Celem zadania jest stworzenie prostej aplikacji generującej raporty pogodowe w postaci pliku .docx na podstawie historycznych danych pogodowych. Dane do wygenerowania raportu mogą być pozyskane z przynajmniej dwóch niezależnych źródeł:

1. Z pliku .xlsx dostępnego lokalnie oraz zawierającego dane w tabeli, np.:

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Wroclaw | Warsaw | Berlin |
| 02.12.2016 | 10 | 8 | 11 |
| xx.xx.xxxx | Yy | Yy | yy |

1. Z serwera OpenWeatherMap przez wykorzystanie udostępnionego API.

## Wskazówki

Funkcjonalność programu możemy łatwo podzielić na podzadania:

1. Moduł prasujący plik .xlsx z pomiarami temperatury - XlsxHistoricalWeatherReader
2. Moduł pobierający dane historyczne ze strony OpenWeatherMap - OWMHistoricalWeatherClient
3. Moduł tworzący raport w pliku .docx - DocxHistoricalWeatherReporter

Jak widać dwa pierwsze moduły realizują bardzo podobną funkcję: są źródłem tych samych danych, które będą przekazywane do modułu DocxHistoricalWeatherReporter.

Dobrą praktyką będzie zaimplementowanie tych modułów tak aby dziedziczyły po wspólnym interfejsie, co obrazuje Rysunek 1.



Rysunek Przykład interfejsu Provider'a dla danych pogodowych oraz jego implementacje dla parsera xlsx oraz klienta OpenWeatherMap API

Po zaimplementowaniu powyższych modułów, można je wykorzystać do stworzenia końcowej aplikacji na przykładzie Rysunku 2.



Rysunek Przykładowa architektura aplikacji

## Działanie aplikacji

Program podczas uruchomienia powinien wczytywać listę miast, dla których chcemy stworzyć raport z danymi historycznymi nt temperatury, np.:

>> python temperatures\_docx\_reporter.py –from=01.12.2016 –to=20.12.2016 –cities=Wroclaw,Berlin,Warsaw

Po wywołaniu, utworzony powinien zostać plik temp\_report.docx o strukturze:

## Wroclaw (heading 2)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| From | To | Average | Max | Min |
| 01.12.2016 | 20.12.2016 | 9.6 | 10 | 8 |

## Berlin (heading 2)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| From |  | To | Average | Max | Min |
| 01.12.2016 |  | 20.12.2016 | 9.8 | 11 | 8 |

## Narzędzia, uwagi

Narzędzia do wykorzystania:

[Python-docx](https://python-docx.readthedocs.io/en/latest/) – tworzenie pliku docx

[OpenPyXL](https://openpyxl.readthedocs.io/en/default/) – parsowanie plików xlsx

[Requests](http://docs.python-requests.org/en/master/) – zapytania http do korzystania z API

[OpenWeatherMap API](https://openweathermap.org/history) – dokumentacja API od OpenWeatherMap (przy korzystaniu z API potrzebny będzie [klucz API, dostępny po darmowej rejestracji](http://openweathermap.org/appid))