

DOKUMENTACJA

„Curling Masters” – Projekt

Dokument przedstawia krótki opis wykonanego przez na projektu.



Katarzyna Macioszek, Ada Majchrzak, Sebastian Janus, Wojciech Strycharczyk, Paweł Kossowski-Skop

Bazy Danych 2021/22

Spis użytych technologii

Przy tworzeniu naszego projektu, zdecydowaliśmy się wykonać większość pracy w języku programowania **Python**. W tym języku wygenerowaliśmy dane, którymi następnie, również w Pythonie, wypełniliśmy bazę danych. Korzystaliśmy przy tym z pakietów takich jak:

* pandas,
* numpy,
* names,
* urllib,
* sqlalchemy,
* random,
* matplotlib,
* datetime,
* dateutil,
* itertools.

Generując dane, poczyniliśmy następujące założenia:

* nasz klub jest klubem amatorskim, do którego zapisać się mogą osoby po 14 roku życia (nie mamy podziału na ligę juniorów i seniorów)
* zawodnicy nie czerpią korzyści finansowych z gry
* rozgrywki odbywają się między drużynami w obrębie klubu, drużyny dopuszczane zostają do rozgrywek rok po osiągnięciu liczby czterech zawodników
* rozgrywki dzielimy na damskie i męskie, rozpoczynają się one w momencie, gdy mamy co najmniej dwie drużyny w danej kategorii i odbywają się co cztery lata dla danej kategorii (płci)
* każdy z zawodników ma swoje buty i nakładki na buty (grippersy), resztę sprzętu zapewnia klub
* musimy mieć zawsze zatrudnionych co najmniej dwóch trenerów, którzy opiekują się wszystkimi zawodnikami w klubie, a do drużyn przydzielani są losowo na czas meczu
* sponsorzy podpisują kontrakty na pełne lata, liczba lat jest generowana losowo
* ID drużyn męskich zaczynają się od 1, natomiast damskich od 100
* ID meczy w rozgrywkach męskich zaczynają się od 10001, a damskie do 20001
* finanse klubu dzielą się na trzy części – wydatki na sprzęt (ID +10000), wypłaty pracownicze (ID +20000) i przychody od sponsorów (ID +30000)
* pierwsza partia zakupionego sprzętu pokryta została ze środków własnych założycieli klubu, a nie z dofinansowania od sponsora

Część związaną z analizą danych i raportem zdecydowaliśmy się wykonać w **Excelu**. Warto jednak wspomnieć, że aby połączyć się z bazą danych MySQL/MariaDB, wymagany jest odpowiedni [konektor](https://dev.mysql.com/downloads/connector/odbc/) ODBC dla MySQL. W zależności od wersji Excela (32 lub 64-bitowa), należy pobrać odpowiednią wersję, zainstalować ją, a następnie dodać konektor z bazą danych *giniewicz.it.* Przy dodawaniu wspomnianego konektora, skorzystaliśmy z [poradnika na YouTube.](https://www.youtube.com/watch?reload=9&v=qK9gPEF606U)

Przy tworzeniu zautomatyzowanego raportu, skorzystaliśmy także z języka **VBA**, który pomógł nam napisać proste makra w pliku Excela.

W tworzeniu diagramu, skorzystaliśmy ze strony **dbdiagram.io**, dzięki której z łatwością mogliśmy przygotować estetyczny schemat naszej bazy danych.

Lista plików i ich zawartości

Nasz projekt zawiera kilka plików.

1. Plik *curling.sql,* który posłużył nam do utworzenia odpowiednich tabel w bazie danych *team7* za pierwszym razem.
2. Skrypt *filler.py*, który odpowiada za generowanie danych i dodawanie ich do tabel na bazie danych. Warto dodać, że kolejne wywołania skryptu nie powodują „podwajania się” danych w tabelach. W pierwszej kolejności dane są usuwane, a następnie dodawane są te, które wygenerowaliśmy.
3. Skoroszyt *Raport.xlsm*, który zawiera prosty interfejs dla użytkownika, przeprowadzone analizy oraz raport. Zgodnie z założeniami projektu, plik ten zapewnia automatyzację generowania raportu.

Kolejność uruchamiania plików

W naszym projekcie, chcieliśmy zapewnić Użytkownikowi **przejrzystość i łatwość obsługi**. Zakładając, że Użytkownik posiada Pythona i wspomniany wcześniej [konektor](#page2) oraz że tabele na bazie danych zostały już wygenerowane, cała funkcjonalność projektu dostępna jest z poziomu pliku Excela.

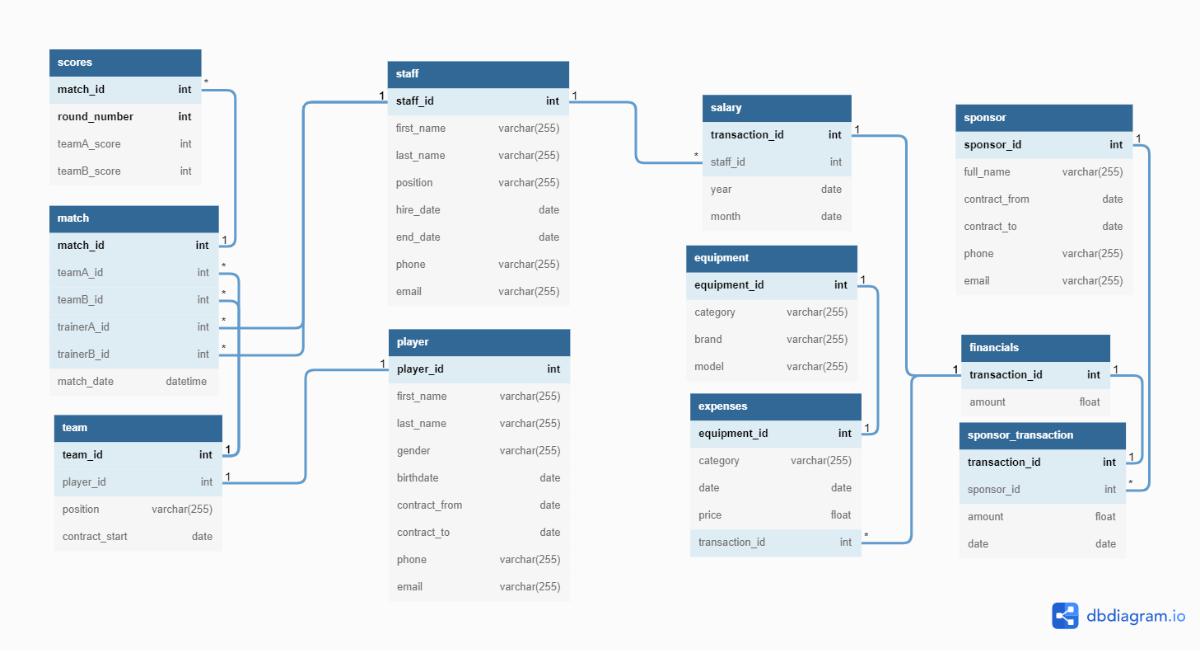


W szarym polu należy wpisać ścieżkę Pythona. Pozwoli to na automatyczne wywołanie skryptu *filler.py* po wciśnięciu przycisku **Generuj nowe dane**, przy założeniu że skrypt oraz plik Excela znajdują się w tym samym folderze.

Przycisk **Odśwież dane** odświeża wszystkie kweredndy (połączenia) z bazą danych w pliku Excela. Pozwala to na odświeżenie raportu, jeśli wygenerujemy nowe dane lub ktoś doda nowe rekordy w bazie danych.

Przycisk **Zapisz raport** pozwala na zapisanie raportu z karty „Raport” w formacie .pdf w wybranym miejscu na komputerze.

Schemat projektu bazy danych



Pogrubione pola oznaczają klucze główne. W przypadku dwóch pogrubionych pól, tabela ma klucz główny składający się z dwóch kolumn.

Relacje

W tej sekcji dokumentacji, opiszemy relacje między tabelami, które stworzyliśmy.

1. Relacja match – scores Relacja 1 do wielu.

Jeden mecz ma wiele wyników, ponieważ w tabeli scores jeden rekord odpowiada jednej rundzie w danym meczu. Nie jest prawdziwa relacja odwrotna, tj. wynik (definiowany przez parę mecz-runda) nie może być przypisany do wielu meczów.

1. Relacja team – match Relacja 1 do wielu.

Jeden zespół może grać w wielu meczach. W tabeli match istnieją pola *teamA\_id* oraz *teamB\_id*, ponieważ mecze prowadzone są pomiędzy drużynami z naszego klubu. Nie jestprawdziwa relacja odwrotna, ponieważ jeden mecz nie może być grany przez kilka drużyn (tj. nie możemy mieć kilku drużyn A i kilku drużyn B w jednym meczu).

1. Relacja team – player Relacja 1 do 1.

Jeden gracz może być pzypisany wyłącznie do jednego zespołu. Warto dodać, że relacja jest 1 do 1, ponieważ nie rozważamy drużyn mikstowych.

1. Relacja staff – match Relacja 1 do wielu.

Relacja jest analogiczna do relacji team – player, tylko zamiast graczy rozpatrujemy trenerów.

1. Relacja staff – salary Relacja 1 do wielu.

Jeden pracownik otrzymuje wiele pensji (każdego miesiąca). Równocześnie, jedna pensja nie może mieć przydzielona kilku pracownikom.

1. Relacja salary – financials Relacja 1 do 1.

Jednej wypłacie przysługuje jeden rekord w tabeli financials (jedna wysokość pensji).

1. Relacja sponsor\_transaction – financials Relacja 1 do 1.

Jednej transakcji od sponsora przysługuje jeden rekord w tabeli financials (jedna wysokość transakcji).

1. Relacja sponsor\_transaction – sponsor Relacja wiele do 1.

Wiele transakcji może pochodzić od jednego sponsora. Jednocześnie, jedna transakcja nie może pochodzić od kilku sponsorów, więc nie jest prawdziwa relacja odwrotna.

1. Relacja expenses – financials Relacja wiele do 1.

Wielu wydatkom przysługuje jeden rekord w tabeli financials (jedna wyskość wydatku). Jest to spowodowane faktem, że dopuszczamy kupowanie kilku rzeczy „za jednym razem”.

1. Relacja expenses – equipment Relacja 1 do 1.

Każda rzecz wyposażenia ma dokładnie jeden rekord w tabeli wydatków.

<EKNF?>

Kilka słów od Autorów

Projektowanie bazy, tworzenie danych do tabel oraz analiza ich zawartości wymagała nie tylko wiedzy i umiejętności, ale również sporo logicznego myślenia. Jedną z najtrudniejszych rzeczy w trakcie pracy nad projektem było generowanie losowych danych. O ile niektóre z nich (przykładowo numery telefonów czy nazwy adresów mailowych) nie stwarzały znacznych problemów na etapie tworzenia, tak wiele innych (generowanie wyników meczów czy tworzenie składu drużyn) wymagało godzin dyskusji i prób, aby uzyskać oczekiwane i zadowalające rezultaty. Poza częścią programistyczną, drugim największym problemem było stworzenie interaktywnego raportu w Excelu tak, aby dane ładowały się automatycznie, zmieniały się wraz z wprowadzonymi zmianami w bazie danych oraz aby umożliwić użytkownikowi wybór interesujących go danych, np. statystyk rozgrywek dla wybranego roku czy drużyny. Ostatecznie większość problemów okazała się możliwa do rozwiązania przy wystarczającym zaangażowaniu zespołu, czego rezultatem jest poprawnie stworzona baza, zadowalający, interaktywny raport oraz dokumentacja, dzięki którym korzystanie z wyników naszej pracy jest proste i przyjemne.