

# Projekt: „Obsługa Pływalni”

Michał Kwarciański

Prowadzący: Mateusz Modrzejewski

## Skrócony opis:

Program obsługujący pływalnię umożliwia rezerwację biletów dla klientów indywidualnych oraz rezerwację torów dla szkółek pływackich z uwzględnieniem dostępności i warunków (w jednym momencie szkółki pływackie nie mogą zarezerwować więcej niż 35% wszystkich torów). W przypadku braku miejsc na wybrany przez klienta termin, program proponuje najbliższy możliwy termin. Program przechowuje historię rezerwacji oraz tworzy dzienny raport finansowy, pokazujący jaki jest przychód pływalni danego dnia. Cennik pływalni jest zróżnicowany, w zależności od pory dnia, dnia tygodnia oraz rodzaju klienta.

## Klasy:

### Czysta logika:

- **SwimmingPool:**

Przyjmuje parametry:

Imię, godziny pracy (lista - [godzina rozpoczęcia, godzina zakończenia]), cennik, liczba torów.

Używa zmiennych: SCHOOLTRACKPARAMETER (maksymalna liczba wszystkich torów rezerwowanych przez szkółki w danej godzinie) i PERSONPERTRACK (maksymalna liczba osób na jednym torze)

Zawarte są w niej mechanizmy działające na torach i miejscach oraz generacja opłaty.

- **IndividualCustomer:**

Przyjmuje parametry:

Imię.

Używa zmiennych: SCHOOLTRACKPARAMETER

Zawiera funkcję obsługującą rezerwację miejsc przez klienta indywidualnego.

- **SwimmingSchool:**

Przyjmuje parametry:

Imię.

Dziedziczy po In IndividualCustomer.

Zawiera funkcję obsługującą rezerwację torów przez szkołkę pływacką.

- **Body:**

Nie przyjmuje parametrów.

Zbiera funkcje, które operują na zmiennych związanych z czasem.

## Instrukcja użytkowania – interfejs w terminalu:

Należy uruchomić plik `interface.py`

Po uruchomieniu w konsoli powinny wypisać się opcje do wyboru. Opcje wybiera się poprzez podanie odpowiedniego numeru (są zawarte w nawiasach kwadratowych).

Opcja „Reserve” daje możliwość kolejnego wyboru -> zarezerwowania miejsc przez klienta indywidualnego lub torów przez szkołkę pływacką.

Opcja „Check booking history” -> Na terminalu wyświetli się historia rezerwacji na pływalni w danym dniu.

Opcja „Check financial raport” -> Na terminalu wyświetli się dochód pływalni w danym dniu.

Opcja „Check parameters of the swimming pool” -> Na terminalu wyświetlą się główne parametry pływalni (nazwa, godziny pracy, liczba torów i miejsc).

Po wybraniu opcji należy postępować zgodnie z instrukcjami zawartymi w dalszych komunikatach programu (z prośbą o podanie dodatkowych danych).

## Instrukcja użytkowania – interfejs graficzny:

Należy uruchomić plik `main_gui.py`

Po uruchomieniu powinno wyświetlić się okienko z zaleceniami o wczytaniu i zapisie danych oraz uproszczona struktura programu.

Żeby **załadować dane z poprzednich uruchomień programu** należy wybrać opcje „Load from file” w zakładce „File”. **Zapis** odbywa się poprzez wybranie „Save to file” w tej samej zakładce.

Żeby **zarezerwować klienta indywidualnego / szkołkę pływacką** należy wcisnąć odpowiedni przycisk w zakładce „Reservation”.

Żeby **sprawdzić historie rezerwacji** lub **dochód pływalni** danego dnia należy wcisnąć odpowiedni przycisk w zakładce „Check”.

Żeby **powrócić do ekranu startowego** należy w zakładce „Start screen” wcisnąć znajdujący się tam przycisk.

Jeśli użytkownik natrafi na sytuację, że o zadanej godzinie nie ma odpowiednio dużo wolnych miejsc, program znajdzie najbliższą datę o zadanej liczbie i wyświetli okienko służące do akceptacji lub odrzucenia nowej daty.

**Przy wyświetlającym się nowym okienku (z wyjątkiem zatwierdzania nowej daty) przed wciśnięciem przycisku „ok” należy wcisnąć przycisk „Accept”, który uruchomi proces sprawdzenia poprawności podanych danych. W przeciwnym wypadku program wyświetli komunikat o tym, że zaszedł błąd przy podawaniu danych!**

Pliki z danymi:

„Swimmingpool\_parametres.py” -> znajdują się w nim dwa modyfikowalne parametry: parametr opisujący, jaki procent torów mogą jednocześnie zajmować szkółki pływackie oraz parametr opisujący limit osób na tor.

Uznałem, że warto te dwie wartości zparametrować w oddzielnym pliku, ponieważ czyni to program bardziej ogólnym.

Istnieją również dwa pliki przechowujące dane pływalni z poprzednich uruchomień programu: „payment\_history.txt” -> zawiera dane dotyczące dochodu pływalni i „booking\_history.txt” -> zawiera dane dotyczące historii rezerwacji na pływalni.

Jest też plik „price\_list.json” -> zawiera on cennik.

Startowe napisy przechowuję w pliku „gui\_start\_text.py”

Refleksje:

Napisanie działającej części logicznej nie sprawiło mi większych problemów. Najtrudniejsze było zaimplementowanie mechanizmu wyszukania nowej daty (a w szczególności przejście na kolejny dzień).

Zrobiłem dodatkowo interfejs graficzny. Było to doświadczenie bardzo rozwijające i poszerzające moją wiedzę. Mam nadzieję, że ta dodatkowa funkcjonalność i poświęcony na nią czas wpłyną pozytywnie na ocenę tego projektu.